ALORA: Affine Low-Rank Approximations

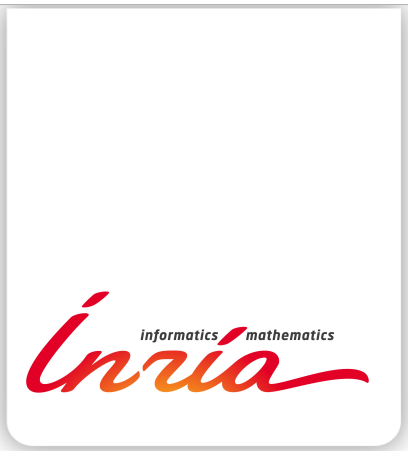
Alan Ayala, Xavier Claeys, Laura Grigori

ALORA: Affine Low-Rank Approximations

Alan Ayala, Xavier Claeys, Laura Grigori

RESEARCH REPORT
N° 9170
April 2018
Project-Team Alpines

ISSN 0249-6399 ISRN INRIA/RR--9170--FR+ENG



ALORA: Affine Low-Rank Approximations

Alan Ayala*†, Xavier Claeys*, Laura Grigori*

Project-Team Alpines

Research Report n° 9170 — April 2018 — 36 pages

Abstract: In this paper we present the concept of affine low-rank approximation for an $m \times n$ matrix, consisting in fitting its columns into an affine subspace of dimension at most $k \ll \min(m, n)$. We show that the optimal affine approximation can be obtained by applying an orthogonal projection to the matrix before constructing its best approximation. Moreover, we present the algorithm ALORA that constructs an affine approximation by slightly modifying the application of any low-rank approximation method. We focus on approximations created with the classical QRCP and subspace iteration algorithms. For the former, we present a detailed analysis of the existing pivoting techniques and furthermore, we provide a bound for the error when an arbitrary pivoting technique is used. For the case of subspace iteration, we prove a result on the convergence of singular vectors, showing a bound that is in agreement with the one for convergence of singular values proved recently. Finally, we present numerical experiences using challenging matrices taken from different fields, showing good performance and validating the theoretical framework.

Key-words: Low rank - QR factorization - subspace iteration - affine subspaces

* INRIA Paris, Sorbonne Université, Univ Paris-Diderot SPC, CNRS, Laboratoire Jacques-Louis Lions, équipe ALPINES, France

† corresponding author: alan.ayala-obregon@inria.fr

RESEARCH CENTRE
PARIS

2 rue Simone Iff
CS 42112 - 75589 Paris Cedex 12

ALORA: Approximations affines de rang faible

Résumé : Dans cet article, nous présentons le concept d'approximation affine de rang faible pour des matrices rectangulaires. Nous montrons comment construire ce type d'approximation en utilisant des projection orthogonaux avec des factorisations QR et itération sur sous-espaces. Nous proposons un algorithme (ALORA) pour calculer une approximation affine de rang faible et le comparons avec des méthodes classiques. Des expériences numériques avec des matrices provenant de différents champs intéressants montrent des bonnes performances et valident le cadre théorique.

Mots-clés : Rang faible, factorisation QR, itération sur sous-espaces, sous-espaces affines

1 Introduction

Many applications in linear algebra, matrix analysis, and statistics require to approximate a given matrix $A \in \mathbb{R}^{m\times n}$ by a rank-k matrix with $k \ll \min(m,n)$. The best approximation can be computed via the singular value decomposition (SVD), however, its computation and storage have $\mathcal{O}(\min(mn^2, m^2n))$ cost for current high accurate routines such as `dgesvj` [9, 10]. Modern attempts to construct faster and accurate low rank approximations have been made using deterministic and randomized algorithms such as QR-based factorizations [21, 7], subspace iteration [20], Monte-Carlo algorithms [15] and random projections [31, 44]. The work by Halko, Martisson and Tropp [22] unifies several randomized approximation methods and presents state-of-the-art algorithms for approximating the SVD.

In this context, standard QR algorithms provide good low-rank approximations and can be created with computational cost of $\mathcal{O}(mnk)$, they have the form

$$A = \sum_{j=1}^{k} q_j r_j^T + E, \tag{1}$$

where $q_j \in \mathbb{R}^m$, $r_j \in \mathbb{R}^n$, $E \in \mathbb{R}^{m\times n}$ is a residual matrix, and k is considered the numerical rank (ϵ-rank) of A when $\|E\|_2 \leq \epsilon$ and ϵ approaches machine epsilon. The classical algorithm for this aim is the QR factorization with column pivoting (QRCP). When k increases, it is known that the theoretical bounds obtained for these algorithms (e.g. $\mathcal{O}(2^k)$ for QRCP) tend to be quite loose in practice (see, for example [7, 19]). Hence, they are widely use for matrix compression and singular values approximation.

In the literature, we can find improved variants of QRCP: on the first hand, methods to reduce the approximation error by improving the choice of the pivoting technique, see e.g. [21], and on the other hand, methods to better approximate the singular values, see e.g. [41]. Of course both approaches can be mixed, and they have in common that they increase the computational cost of QRCP, and can be considerably more expensive when dealing with large matrices. In this context, we present an algorithm named *ALORA* that can be adapted to any low-rank approximation method, and can improve their approximation properties by simply adding few computations that are a lot less expensive compared with the cost of the algorithm itself.

In order to elaborate faster (ideally linear time cost) algorithms, we have to exploit the matrix structure. For instance, when A is a sparse matrix, the `PROPACK` [28] and `ARPACK` [29] softwares can compute a sparse approximation of the SVD based on the Lanczos algorithm with a much smaller computational cost than the SVD. On the other hand, if A is a dense matrix, one case that allows to exploit its structure is when each of its entries are constructed as $A_{ij} = f(x_i, y_j)$ where $\Gamma_S = \{x_1 \cdots x_m\}$ and $\Gamma_T = \{y_1, \cdots, y_m\}$ are two sets of pairwise distinct points in $\mathbb{R}^d$, with $d = 1, 2$ or 3, and $f : \Gamma_S \times \Gamma_T \to \mathbb{R}$ admits a decomposition by *functional skeletons* [1] of the type

$$f(x,y) = \sum_{j=0}^{k} g(x)h(y) + E_k(x,y), \tag{2}$$

where $\|E_k(x,y)\|_2 \leq \epsilon_k$ and $\epsilon_k \to 0$ when $k \to \infty$. From (2), it is clear that A can be approximated by a rank-k matrix and k is referred to as its numerical rank whenever ϵ_k is close to the machine epsilon. Such matrices arise when solving integral equations in the framework of the Boundary Element Method (BEM), and they are called admissible submatrices in the context of hierarchical matrices. It is known that by choosing Γ_S and Γ_T, e.g. using a hierarchical partition, the singular values of these kind of matrices decrease exponentially [2]. There exist a wide list of algorithms, among them, we mention two that are representative of different approaches and hence can show the pros and cons of the algorithms of their kind, the Adaptive Cross Approximation (ACA) [1, 2] and the Black Box Fast Multipole Method (BBFMM) [14]. Both, ACA and BBFMM allow to compute a low-rank approximation of a BEM matrix

with linear computational cost. The ACA algorithm relies on approximating the maximum volume submatrix of A and it is widely use in practice. However, it is known that its approximation error can get large [2, Sec. 3.4.3]. On the other hand, BBFMM is one of the many kernel independent approaches that work well in practice, however, as most of them, it has restrictions on its use, for instance BBFMM works only for kernels that are non-oscillatory. To avoid the issues of the two previously mentioned methods, one can construct a purely algebraic approach using only the entries of the matrix, this can be done with a QR-based approximation such as the one proposed by the IE-QR algorithm [35] which constructs a low-rank QR approximation using the modified Gram-Schmidt algorithm. However, even if it provides good results for matrices constructed with carefully selected pairs of interaction domains Γ_S and Γ_T, its stability is not guaranteed and it costs $\mathcal{O}((\max(m,n))^{\frac{3}{2}})$. In this context, using tools from statistics, we first define the correlation for a matrix by means of a correlation vector and a correlation coefficient, and further we show that matrices with exponentially decreasing singular values, e.g. BEM matrices, tend to have high correlation and how to exploit this feature. We provide an algorithm named AGC that works well in practice for these kind of matrices. However, AGC has complexity $\mathcal{O}(mnk)$, which is not desirable in practice. Currently the authors work on the construction of an accurate linear-cost approximation method for these kind of matrices.

Theoretical and algorithmic contributions.

In this article we present a new approach to construct low-rank approximations using projection techniques into an affine subspace. This is, we approximate $A \in \mathbb{R}^{m\times n}$ as

$$A \approx \xi_k := \big(\sum_{j=1}^{k-1} q_j q_j^T\big) A(I - zz^T) + (Az)z^T, \tag{3}$$

where $(Az)z^T$ can be seen as a translation matrix. We geometrically explore the construction of approximation (3) using QR factorization, based on Householder reflections, as well as subspace iteration. We provide an algorithm referred to as ALORA that can be adapted to any low-rank approximation method. We apply the ALORA algorithm on a set of challenging matrices used in previous related papers and discuss the cases where this technique improves the approximation error. In addition to the ALORA algorithm we provide a heuristic algorithm named AGC, envisaged for matrices with exponentially decreasing singular values, which can be used to construct faster approximations and estimate the matrix norm.

We also present a survey of the different techniques to construct a QR based low-rank approximation, providing a bound when a general pivoting technique is used. Furthermore, we also prove the convergence of singular vectors for the subspace iteration algorithm. And finally, we provide some insights that allow to envisage linear cost approximations for BEM matrices.

The article is organized as follows. Section 2 presents classical methods to compute a low-rank factorization by means of QR factorization, subspace iteration and their randomized versions. We analyze and compare the different techniques employed by state-of-the-art algorithms. Section 3 presents the concept of affine low-rank approximation, it starts by analyzing a general framework for constructing the approximation by using projections of rows and columns. It presents the problem of finding the optimal Householder reflectors and solves it by using the total least squares technique, the analysis from this section leads to the construction of the ALORA algorithm. Next, in Section 4 we analyze matrices for which an affine approximation would be advantageous, and we also define a correlation coefficient for any real matrix using statistical tools. We show that matrices with exponentially decreasing singular values, in particular BEM matrices, have high correlation coefficient, and a heuristic algorithm named AGC is developed to approximate them. Moreover, we also provide a simple, but accurate, approximation of the spectral norm. Section 5 presents and discusses several numerical experiments to validate the algorithms ALORA and AGC by using a set of challenging matrices arising from different interesting fields. Finally, Section 6 concludes our paper.

2 Definitions and Background

2.1 Notations

Let us first state notational conventions that we shall use all through this article. In the sequel, $A \in \mathbb{R}^{m\times n}$ refers to a (not necessarily square $m \neq n$) real matrix. We denote $\|A\|_2$ and $\|A\|_F$ the spectral and Frobenius norms respectively and $\|A\|_{\max} := \max_{i,j}|A_{i,j}|$ is the Chebyshev (or maximum) norm. We use `MATLAB` notation to present some matrix operations.

Remark 2.1. The results from this paper can be extended to rectangular complex matrices, by making small appropriate changes in the definitions, statements and proofs.

When given two matrices $W_1, W_2 \in \mathbb{R}^{m\times k}$ with orthonormal columns, let $S_i = \mathrm{ran}(W_i)$, for $i = 1, 2$, refer to the vector subspace spanned by the columns of W_i, then $\angle(S_1, S_2) := \arcsin(\|W_1W_1^T - W_2W_2^T\|_2)$ refers to the angle between these two spaces.

2.2 Best Low-rank Approximation

For any matrix $A \in \mathbb{R}^{m\times n}$, there exists $\Sigma \in \mathbb{R}^{m\times n}$ and two orthogonal matrices $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ such that

$$A = U\Sigma V^T, \tag{4}$$

where

$$\begin{cases} \Sigma_{jj} = \sigma_j & \text{for } j = 1, \cdots, \min(m, n), \\ \Sigma_{ij} = 0 & \text{elsewhere.} \end{cases}$$

The values σ_j are known as singular values and we assume a non-increasing ordering $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\min(m,n)} \geq 0$, so that Σ is uniquely determined by A, cf. [24, Thm. 3.1.1]. The right and left singular vectors are defined, respectively, as the columns of the matrices U and V.

For any given matrix $M \in \mathbb{R}^{m\times n}$, we denote its singular triplets as $(u_j(M), v_j(M), \sigma_j(M))$, or simply (u_j, v_j, σ_j) when this is clear from the context, where u_j and v_j are the left and right singular vectors corresponding to the singular value σ_j.

Definition 2.2. The rank of a matrix $A \in \mathbb{R}^{m\times n}$ is defined as the smallest integer i for which $\sigma_{i+1} = 0$, we use the notation $r := \mathrm{rank}(A)$.

Remark 2.3. Along this paper we consider $\mathrm{rank}(A) \geq k$, since we are interested on obtaining a rank-k approximation of A.

Next, let us introduce the truncated SVD of A, which is a rank-k approximation, defined as

$$A_k := U_k\Sigma_kV_k^T \equiv \sum_{i=1}^{k} u_i\sigma_iv_i^T, \tag{5}$$

where $U_k := [u_1, \cdots, u_k]$, $\Sigma_k := \mathrm{diag}(\sigma_1, \ldots, \sigma_k)$ and $V_k := [v_1, \cdots, v_k]$. For the spectral and Frobenius norms, a fast algebraic calculus shows that

$$\|A_k - A\|_2 = \sigma_{k+1}, \qquad \|A_k - A\|_F = \sqrt{\sigma_{k+1}^2 + \cdots + \sigma_r^2}.$$

The following theorem states that the truncated SVD is the best low-rank approximation for any unitarily invariant norm, cf. Mirsky [33] and Eckart and Young [12].

Theorem 2.4. *(Mirsky, [33, Thm. 2]) Consider the matrix $A \in \mathbb{R}^{m\times n}$, with singular triplets (u_i, v_i, σ_i) for $i = 1, \cdots, \min(m, n)$. Then, $A_k = \sum_{i=1}^{k} u_i\sigma_i v_i^T$ is a solution of the following problem*

$$\begin{cases} \text{Find } B \in \mathbb{R}^{m\times n} \text{ of rank at most } k, \text{ such that} \\ \quad \|A - B\| \leq \|A - C\|, \quad \forall\, C \in \mathbb{R}^{m\times n} \text{ of rank at most } k, \end{cases} \tag{6}$$

where $\|\cdot\|$ stands for any unitarily invariant norm.

Remark 2.5. Note that problem (6) has a unique solution when the Frobenius norm is used if and only if $\sigma_k \neq \sigma_{k+1}$, cf. [12]. If the spectral norm is used then, as explained in [20], the solution of problem (6) is not unique. For instance, for any $0 \leq \theta \leq 1$ the matrix $B = A_k - \theta\sigma_{k+1}U_kV_k^T$ is a solution.

The following theorem presents some useful inequalities that will be helpful in next sections.

Theorem 2.6. *(Horn and Jonhson, [24, Thm. 3.3.16]) Let $A, B \in \mathbb{R}^{m\times n}$ and $q = \min(m, n)$ then*

$$\sigma_{i+j-1}(AB^T) \leq \sigma_i(A)\sigma_j(B), \tag{7}$$

and

$$\sigma_{i+j-1}(A + B) \leq \sigma_i(A) + \sigma_j(B), \tag{8}$$

holds for $1 \leq i, j$ and $i + j \leq q + 1$.

2.3 Low-Rank Approximation using Pivoted QR Factorization

We construct the low-rank QR factorization using Householder reflectors, we choose it over the classical Gram-Schmidt orthogonalization since the former has better stability [6, Sec. 3.4].

Definition 2.7 (Householder reflector, cf. [25])**.** Given $u \in \mathbb{R}^j$, the Householder reflector associated to u is a linear transformation that describes a reflection about an hyperplane orthogonal to u and passing through the origin. Its corresponding matrix is referred to as the Householder matrix

$$H := I - \frac{2}{\|v\|_2^2}vv^T \in \mathbb{R}^{j\times j}, \tag{9}$$

where $v = u - \|u\|_2 e_1$ is known as the Householder vector and $e_1 = (1, 0, \cdots, 0)^T \in \mathbb{R}^j$. Note that H is a symmetric orthogonal matrix, and it holds $He_1 = u$ and $Hu = \|u\|_2 e_1$.

A complete pivoted QR factorization can be constructed by applying n Householder reflections to the columns of A [17, Ch. 5]. Let us present this factorization inductively. For any $k = 1, \cdots, n$, the k-th step of the factorization, i.e. applying the first k reflections, has the form

$$\tilde{Q}_k \cdots \tilde{Q}_2\tilde{Q}_1 A(:, p_k) = R_k = \begin{array}{c} \\ k \\ m-k \end{array} \begin{array}{c} \begin{array}{cc} k & n-k \end{array} \\ \begin{bmatrix} R_{11}^{(k)} & R_{12}^{(k)} \\ 0 & R_{22}^{(k)} \end{bmatrix} \end{array}, \tag{10}$$

where p_k is a permutation vector that interchanges the columns $A(:, j)$ and $A(:, p(j))$, for $j = 1, \cdots, k$. The matrix $R_{11}^{(k)}$ is upper triangular, and

$$\tilde{Q}_1 := H_1 \quad \text{and} \quad \tilde{Q}_j := \begin{bmatrix} I_{j-1} & 0 \\ 0 & H_j \end{bmatrix} \quad \text{for } 2 \leq j \leq k,$$

are $m \times m$ matrices, where H_1 is the Householder matrix corresponding to the $p(1)$-th column of A, and for $2 \leq j \leq k$ we denote the identity matrix of size $(j-1) \times (j-1)$ as I_{j-1}, and H_j is the Householder

matrix corresponding to the $p(j)$-th column of $R_{22}^{(j-1)}$. Hence, the matrices $\tilde{Q}_j$ are symmetric orthogonal matrices, and we define

$$Q = \begin{array}{c} \\ m \end{array} \begin{array}{c} \begin{array}{cc} k & m-k \end{array} \\ \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \end{array} := \tilde{Q}_k \cdots \tilde{Q}_2 \tilde{Q}_1 \quad \text{and} \quad P_k := I(:, p_k),$$

where $I \in \mathbb{R}^{n \times n}$ is the identity matrix, so that $AP_k = QR_k$ is known as the truncated QR factorization of A.

A rank-k QR approximation directly follows by rewriting (10) as

$$\begin{aligned} A &= \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \begin{bmatrix} R_{11}^{(k)} & R_{12}^{(k)} \\ 0 & R_{22}^{(k)} \end{bmatrix} P_k^T \\ &= \underbrace{Q_1 \begin{bmatrix} R_{11}^{(k)} & R_{12}^{(k)} \end{bmatrix} P_k^T}_{=:\xi_k} + \underbrace{Q_2 \begin{bmatrix} 0 & R_{22}^{(k)} \end{bmatrix} P_k^T}_{=:E_k}. \end{aligned} \tag{11}$$

The matrix ξ_k is the rank-k QR approximation of A and the approximation error (for any unitarily invariant norm $\|\cdot\|$) is given by

$$\|A - \xi_k\| = \|E_k\| = \|Q_2 [0 \quad R_{22}^{(k)}] P_k^T\| = \|[0 \quad R_{22}^{(k)}]\| = \|R_{22}^{(k)}\|. \tag{12}$$

Computing ξ_k is typically much faster than computing the truncated SVD. The accuracy of the approximation greatly depends on the selected permutation P_k, and hence we analyze this choice in the next subsection.

2.4 Choosing a Permutation for a QR Factorization

The choice of the permutation is of great importance to control the error of a low-rank QR approximation, below we summarize state-of-the-art techniques for this purpose.

Choosing the permutation using the maximal volume criterium

The following theorem states that we can find permutations such that the error, in the maximum norm, will be of the same order as σ_{k+1}.

Theorem 2.8. *(Goreinov et al., [18, Thm. 2.1]) Let us consider the matrix*

$$\bar{A} = \begin{bmatrix} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{bmatrix},$$

where $\bar{A}_{11} \in \mathbb{R}^{k \times k}$ has maximal volume (i.e., maximum determinant in absolute value) among all $k \times k$ submatrices of $\bar{A}$. Then,

$$\|S(\bar{A}_{11})\|_{\max} \leq (k+1)\sigma_{k+1}(\bar{A}), \tag{13}$$

where $S(\bar{A}_{11}) = \bar{A}_{22} - \bar{A}_{21}\bar{A}_{11}^{-1}\bar{A}_{12}$.

Let us apply the previous theorem at the step k of a truncated QR factorization of type (11), in this case we need to use two permutations P_r and P_c, this is

$$\bar{A} = P_r A P_c = \begin{array}{c} \\ k \\ m-k \end{array} \begin{array}{c} \begin{array}{cc} k & n-k \end{array} \\ \begin{bmatrix} \bar{A}_{11} & \bar{A}_{12} \\ \bar{A}_{21} & \bar{A}_{22} \end{bmatrix} \end{array} = \begin{array}{c} \\ k \\ m-k \end{array} \begin{array}{c} \begin{array}{cc} k & m-k \end{array} \\ \begin{bmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{bmatrix} \end{array} \begin{array}{c} \\ k \\ m-k \end{array} \begin{array}{c} \begin{array}{cc} k & n-k \end{array} \\ \begin{bmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{bmatrix} \end{array},$$

where the row and column permutations P_r and P_c are obtained such that the submatrix $\bar{A}_{11}$ has maximal volume among all $k \times k$ matrices of $\bar{A}$. Next, a direct calculus shows that $S(\bar{A}_{11}) = S(Q_{11})R_{22}$, with

$$S(Q_{11}) := Q_{22} - Q_{21}Q_{11}^{-1}Q_{12} = Q_{22}^{-T},$$

where the last equality can be verified by computing $Q_{22}^T S(Q_{11})$ or it can also be found in [36, proof of Thm. 3.7]. Hence, the approximation error is given by

$$\|R_{22}\|_2 \leq \|Q_{22}^T S(\bar{A}_{11})\|_2 \leq \|S(\bar{A}_{11})\|_2 \leq (m-k)\|S(\bar{A}_{11})\|_{\max} \leq (m-k)(k+1)\sigma_{k+1}(A), \tag{14}$$

where we have used the facts that $\sigma_{k+1}(A) = \sigma_{k+1}(\bar{A})$, $\|Q_{22}\|_2 \leq 1$, since it is a submatrix of an orthogonal matrix, and for $M \in \mathbb{R}^{m\times n}$ it holds $\|M\|_2 \leq \sqrt{mn}\|M\|_{\max}$.

Even though the bound (14) is very good, in practice finding a submatrix of maximum volume has been proven to be NP-hard [5].

Choosing the permutation using classical column pivoting

The classical QR with column pivoting [17, Alg. 5.4.1], which we refer to as QRCP, computes a rank-k approximation as in equation (11), where the permutation $P_k = I(:, p_k)$ is constructed such the $p_k(1)$-th column of A is the one with largest norm, and for $2 \leq j \leq k$ it holds that the $p_k(j)$-th column of R_{22}^{j-1} is the one of largest norm. This is a greedy approach to maximize the volume of the factor $R_{11}^{(k)}$. Stopping the algorithm at step k, it produces a rank-k QR approximation where the matrix $Q_1 R_{11}^{(k)}$ is a set of k columns of A. The approximation error is given by [21, Thm. 7.2]

$$\|R_{22}^{(k)}\|_2 \leq 2^k \sqrt{n-k}\,\sigma_{k+1}. \tag{15}$$

This exponential bound is typically pessimistic compared to what is observed in practice, and the cost of the algorithm is $\mathcal{O}(mnk)$.

Other techniques to choose the permutation

Different authors have proposed algorithms to reduce the exponential bound on QRCP to polynomial bounds, in general

$$\|R_{22}^{(k)}\|_2 \leq f(k,n)\,\sigma_{k+1}, \tag{16}$$

where $f(k,n)$ is a function on k and n, see e.g. [7, 21, 37]. For a compilation of some of the different algorithms of this kind and their computational complexity see [4, Table 1].

For example, the strong rank revealing factorization of [21] produces a rank-k QR approximation with error

$$\|R_{22}^{(k)}\|_2 \leq \sqrt{1+\nu^2 k(n-k)}\,\sigma_{k+1}, \tag{17}$$

where $\nu > 1$ is a constant, cf. [21, Thm. 3.2]. This algorithm costs $\mathcal{O}(kmn\log_\nu(n))$. Note that, if $m \geq n$, the choice $\nu = 1$ gives a better bound on the error than the one from (14). However, for this case the algorithm might also have exponential cost to compute the approximation.

Approximation error for an arbitrary permutation

Now we analyze the error of approximation using a low-rank QR approximation with an arbitrary permutation P. Consider the truncated QR factorization of A,

$$AP = QR = \begin{array}{c} \\ m \end{array} \begin{array}{c} \begin{array}{cc} k & m-k \end{array} \\ \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \end{array} \begin{array}{c} \begin{array}{c} k \\ m-k \end{array} \\ \begin{bmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{bmatrix} \end{array} \begin{array}{c} \begin{array}{cc} k & n-k \end{array} \\ \\ \end{array}. \tag{18}$$

Next, note that $Q_1 R_{11} = AP(:, 1:k)$ and that the error of a QR approximation given in (12) can also be obtained as

$$\|R_{22}\|_2 = \|(I - Q_1 Q_1^T)A\|_2, \tag{19}$$

where I is the identity matrix and $Q_1 Q_1^T$ is the orthogonal projector over the subspace generated by the first k columns of AP. This is true since

$$\|(I - Q_1 Q_1^T)A\|_2 = \|(Q^T - Q^T Q_1 Q_1^T)QRP^T\|_2 = \|R - Q^T Q_1 Q_1^T QR\|_2,$$

and,

$$Q^T Q_1 Q_1^T Q = \begin{bmatrix} I \\ Q_2^T Q_1 \end{bmatrix} \begin{bmatrix} I & Q_1^T Q_2 \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix}, \tag{20}$$

which holds since the columns of Q_1 and Q_2 are mutually orthogonal.

Note that from the previous analysis, a simple bound can be obtained for the error using a general permutation, this is

$$\|R_{22}\|_{\max} \leq \|R_{22}\|_2 = \|(I - Q_1 Q_1^T)A\|_2 \leq \|I - Q_1 Q_1^T\|_2 \|A\|_2 \leq \|A\|_2 \leq \sqrt{mn}\|A\|_{\max}. \tag{21}$$

The following lemma, using an assumption on the right singular vectors, provides a bound of type (16) for the approximation error when using an arbitrary permutation to compute a low-rank QR approximation.

Lemma 2.9. Let $A \in \mathbb{R}^{m \times n}$, consider its truncated QR factorization,

$$AP = QR = \begin{array}{c} \\ m \end{array} \begin{array}{c} \begin{array}{cc} k & m-k \end{array} \\ \begin{bmatrix} Q_1 & Q_2 \end{bmatrix} \end{array} \begin{array}{c} \begin{array}{c} k \\ m-k \end{array} \\ \begin{bmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{bmatrix} \end{array} \begin{array}{c} \begin{array}{cc} k & n-k \end{array} \\ \\ \end{array}, \tag{22}$$

where $P \in \mathbb{R}^{n \times n}$ is an arbitrary permutation. Define

$$\begin{bmatrix} \Omega_1 \\ \Omega_2 \end{bmatrix} := (V^T P)(:, 1:k), \tag{23}$$

where $V \in \mathbb{R}^{n \times n}$ is the matrix containing the right singular vectors of A, as defined in (4). Assuming that Ω_1 is non-singular, then

$$\|R_{22}\|_2 \leq \sqrt{1 + \|\Omega_2 \Omega_1^{-1}\|_2^2}\, \sigma_{k+1}(A). \tag{24}$$

Proof. Consider the matrix $\Omega \in \mathbb{R}^{n \times k}$ given as

$$\Omega = \begin{bmatrix} I_k \\ 0 \end{bmatrix},$$

where I_k denotes the identity matrix of size $k \times k$. Next, consider the SVD decomposition $A = U\Sigma V^T$. Define $\tilde{V}^T = V^T P$ and the matrices

$$\bar{A} := AP = U\Sigma\tilde{V}^T, \qquad \text{and} \qquad Y := AP\Omega = U\Sigma\tilde{V}^T(:, 1:k) = U\Sigma \begin{bmatrix} \Omega_1 \\ \Omega_2 \end{bmatrix}. \tag{25}$$

Next, note that Y is the matrix consisting of the first k columns of $\bar{A}$, and its orthogonal projector is $Q_1 Q_1^T$. Then, as showed in (19) we have

$$\|R_{22}\|_2 = \|(I - Q_1 Q_1^T)A\|_2. \tag{26}$$

Finally, by applying [22, Thm. 9.1] on $\bar{A}$, we get

$$\|(I - Q_1 Q_1^T)A\|_2 = \|(I - Q_1 Q_1^T)\bar{A}\|_2 \leq \sqrt{1 + \|\Omega_2 \Omega_1^{-1}\|_2^2}\, \sigma_{k+1}(A). \tag{27}$$

□ □

2.5 Low-rank Approximation using Subspace Iteration

Several algorithms have been developed in order to reduce the computational cost and error of the approximation. Among them, methods based on subspace iteration [17, Ch. 7, 8] have been shown to produce a good rank-k approximation with cost between $\mathcal{O}(mn\log(k))$ and $\mathcal{O}(mnk)$, see for example [11, 22, 32].

Algorithm 2.1 presents the basic subspace iteration, this algorithm is well known in the literature and versions of it have been presented by different authors, see for example [20, 22]. It takes as input an $m \times n$ matrix A, a small integer q (that is usually taken as $q = 1$ or $q = 2$), and a matrix $\Omega \in \mathbb{R}^{n\times l}$ that a priori can be chosen deterministically or randomly, and such that the span of columns of $A\Omega$ is as close as possible to the span of columns of A.

Algorithm 2.1 $[\xi_k]$ = SSITER(A,Ω,k,q)

Requires: $A \in \mathbb{R}^{m\times n}$, $\Omega \in \mathbb{R}^{n\times l}$, with $l \geq k$.
Returns: rank-k approximation of A.
1: Perform $Y = (AA^T)^q A\Omega$.
2: Compute the (economic) QR decomposition $Y = QR$.
3: Form $B = Q^T A$.
4: Find B_k, the rank-k truncated SVD of B.
5: Set $\xi_k := QB_k$.

In line 2 of the algorithm, an economic QR factorization means that we take $Y = QR$ with $Q \in \mathbb{R}^{m\times t}$ and $R \in \mathbb{R}^{t\times n}$, where $t = \min(m, n)$. For numerical stability, the matrix Y in line 1 should be computed as in [20, Alg. A.1].

Note that Algorithm 2.1 could stop at line 3 and return the rank-l matrix $QQ^T A$ as the the low-rank approximation of A, indeed it is known in the literature (see e.g. [22]) that for any matrix $B \in \mathbb{R}^{l\times n}$, it holds $\|A - QQ^T A\|_2 \leq \|A - QB\|_2$. Then, $\|A - QQ^T A\|_2 \leq \|A - \xi_k\|_2$. Hence, computing ξ_k provides a less accurate low-rank approximation than $QQ^T A$, in terms of the norm of the approximation error. However, obtaining ξ_k can provide better approximation of the singular values [20]. In Theorem 2.10, we prove that the first k columns of Q converge to the first k left singular vectors of A at an exponential rate.

Considering that we chose the approximation ξ_k from Algorithm 2.1 for the matrix $A = U\Sigma V^T$ with singular values $\sigma_1, \cdots, \sigma_r$. It is possible to obtain rapidly converging approximations of the matrix and its singular values, provided that the matrix $\widehat{\Omega}$ defined as

$$\widehat{\Omega} := V^T\Omega = \begin{matrix} l-p \\ n-l+p \end{matrix} \begin{bmatrix} \widehat{\Omega}_1 \\ \widehat{\Omega}_2 \end{bmatrix}, \qquad 0 \leq p \leq l-k, \tag{28}$$

where p is known as oversampling parameter, is such that its submatrix $\widehat{\Omega}_1$ is full row rank. In fact, we have the bounds (cf. [20, Thms. 4.3 and 4.4]),

$$\sigma_j \geq \sigma_j(B_k) \geq \frac{\sigma_j}{\sqrt{1+\psi^2\|\widehat{\Omega}_2\|_2^2\,\|\widehat{\Omega}_1^\dagger\|_2^2}} \tag{29}$$

and,

$$\|A-\xi_k\|_2 \leq \sqrt{\sigma_{k+1}^2+\omega^2\|\widehat{\Omega}_2\|_2^2\,\|\widehat{\Omega}_1^\dagger\|_2^2}, \tag{30}$$

where $\psi = \left(\frac{\sigma_{l-p+1}}{\sigma_j}\right)^{2q+1}$, $\omega = \sqrt{k}\sigma_{l-p+1}\left(\frac{\sigma_{l-p+1}}{\sigma_k}\right)^{2q}$, $0 \leq p \leq l-k$ and $\widehat{\Omega}_1\widehat{\Omega}_1^\dagger = I$.

A randomized version of Algorithm 2.1 can also be obtained by letting Ω be a Gaussian matrix, meaning that its entries are independent standard normal variables of unit-variance and zero mean. The matrix $\widehat{\Omega}_1$ as in (28) is still a Gaussian matrix [22], and it is proven that if $l-p \geq 2$ then $\widehat{\Omega}_1$ has full rank with probability 1, [20, Lem. 5.2]. By setting $l = 2k$, $q = 0$, and Ω as a Gaussian matrix, Algorithm 2.1 produces a rank-k approximation with expected error [22, Thm.1.2],

$$\mathbb{E}\|A-\xi_k\|_2 \leq \left(2+4\sqrt{\frac{2\min(m,n)}{k-1}}\right)\sigma_{k+1}. \tag{31}$$

Algorithm 2.1 works very well in practice and has computational complexity of $\mathcal{O}(mnk)$. In the next section we construct approximations of the matrix A using methods described in this section to approximate the left singular vectors which turns out to be extremely important for our analysis. In this context, the following theorem proves a result for the convergence of singular vectors when using Algorithm 1.

Theorem 2.10. *Consider $\Omega \in \mathbb{R}^{m\times l}$ and $A \in \mathbb{R}^{m\times n}$, with SVD decomposition $A = U\Sigma V^T$. Consider the QR factorization $QR = (AA^T)^qA\Omega$ and let $Q_k = [q_1, \cdots, q_k]$ and $U_k = [u_1, \cdots, u_k]$ be matrices constructed with the first k columns of Q and U respectively. Considering the partition*

$$V^T\Omega := \begin{matrix} \\ k \\ n-k \end{matrix} \begin{matrix} k & l-k \\ \end{matrix} \begin{bmatrix} \Omega_\alpha & Z_1 \\ \Omega_\beta & Z_2 \end{bmatrix}, \tag{32}$$

if Ω_α is invertible, defining $\varphi = \angle(\mathrm{ran}(Q_k), \mathrm{ran}(U_k))$, then

$$\sin(\varphi) \leq \left(\frac{\sigma_{k+1}}{\sigma_k}\right)^{2q+1}\|\Omega_\beta\Omega_\alpha^{-1}\|_2.$$

Proof. First, consider the partitions

$$\Sigma = \begin{array}{c} \\ k \\ n-k \end{array} \begin{array}{c} \begin{array}{cc} k & n-k \end{array} \\ \begin{bmatrix} D_k & 0 \\ 0 & D_s \end{bmatrix} \end{array}, \quad U = \begin{array}{c} \\ m \end{array} \begin{array}{c} \begin{array}{cc} k & n-k \end{array} \\ \begin{bmatrix} U_k & U_s \end{bmatrix} \end{array}, \quad Q = \begin{array}{c} \\ m \end{array} \begin{array}{c} \begin{array}{cc} k & n-k \end{array} \\ \begin{bmatrix} Q_k & Q_s \end{bmatrix} \end{array}. \tag{33}$$

Next, analyzing the QR factorization we get

$$(AA^T)^q A\Omega = U\Sigma^{2q+1}V^T\Omega = QR = \begin{bmatrix} Q_k & Q_s \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{bmatrix}, \tag{34}$$

where $R_{11} \in \mathbb{R}^{k\times k}$. Hence,

$$\begin{bmatrix} U_k & U_s \end{bmatrix} \begin{bmatrix} D_k & 0 \\ 0 & D_s \end{bmatrix}^{2q+1} \begin{bmatrix} \Omega_\alpha & Z_1 \\ \Omega_\beta & Z_2 \end{bmatrix} = \begin{bmatrix} Q_k & Q_s \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} \\ 0 & R_{22} \end{bmatrix}. \tag{35}$$

Comparing the first k columns of both sides of equation (35), we get an embedded QR factorization of a matrix W defined as

$$W := \begin{bmatrix} U_k & U_s \end{bmatrix} \begin{bmatrix} D_k & 0 \\ 0 & D_s \end{bmatrix}^{2q+1} \begin{bmatrix} \Omega_\alpha \\ \Omega_\beta \end{bmatrix} = Q_k R_{11}. \tag{36}$$

Next, note that we search $\sin(\varphi) = \|U_k U_k^T - Q_k Q_k^T\|_2$, which by [17, Thm. 2.6.1] is equivalent to

$$\sin(\varphi) = \|U_s^T Q_k\|_2. \tag{37}$$

Define the matrix

$$X := \Omega_\alpha^{-1} D_k^{-(2q+1)} \in \mathbb{R}^{k\times k}, \tag{38}$$

which is non-singular by assumption of the theorem, and consider the QR factorization of WX,

$$WX = U \begin{bmatrix} D_k & 0 \\ 0 & D_s \end{bmatrix}^{2q+1} \begin{bmatrix} \Omega_\alpha \\ \Omega_\beta \end{bmatrix} X = \tilde{Q}_k \tilde{R}_{11}. \tag{39}$$

where $\tilde{R}_{11} \in \mathbb{R}^{k\times k}$. Replacing (38) on (39), we get

$$\begin{bmatrix} I_{k\times k} \\ D_s^{(2q+1)} \Omega_\beta \Omega_\alpha^{-1} D_k^{-(2q+1)} \end{bmatrix} = U^T \tilde{Q}_k \tilde{R}_{11} = \begin{bmatrix} U_k^T \tilde{Q}_k \\ U_s^T \tilde{Q}_k \end{bmatrix} \tilde{R}_{11}, \tag{40}$$

from which we deduce that $\tilde{R}_{11}^{-1} = U_k^T \tilde{Q}_k$. Next, let us compute

$$\|U_k U_k^T - \tilde{Q}_k \tilde{Q}_k^T\|_2 \equiv \|U_s^T \tilde{Q}_k\|_2 = \|D_s^{(2q+1)} \Omega_\beta \Omega_\alpha^{-1} D_k^{-(2q+1)} \tilde{R}_{11}^{-1}\|_2. \tag{41}$$

Equation (41) is important since by [20, Lem. 4.1] we have that factorizations (36) and (39) have the property

$$Q_k Q_k^T = \tilde{Q}_k \tilde{Q}_k^T. \tag{42}$$

Finally, from (41), (42) and the fact that $\|\tilde{R}_{11}^{-1}\|_2 = \|U_k^T \tilde{Q}_k\|_2 \leq 1$, we obtain

$$\sin(\varphi) = \|U_k U_k^T - Q_k Q_k^T\|_2 = \|U_k U_k^T - \tilde{Q}_k \tilde{Q}_k^T\|_2 \leq \|D_s^{(2q+1)} \Omega_\beta \Omega_\alpha^{-1} D_k^{-(2q+1)}\|_2. \tag{43}$$

Hence,

$$\sin(\varphi) \leq \left(\frac{\sigma_{k+1}}{\sigma_k}\right)^{2q+1} \|\Omega_\beta \Omega_\alpha^{-1}\|_2. \tag{44}$$

□ □

The previous theorem shows that the subspace generated by the span of the k first columns of Q (obtained by Algorithm 2.1 applied to a matrix $A \in \mathbb{R}^{m\times n}$) converges with an exponential rate to the subspace generated by the first k left singular vectors. This agrees with the exponential rates obtained for the convergence of singular vectors and approximation error (29) and (30), and was predicted in a previous work [20, Sec. 9].

Remark 2.11. When Ω_α and Ω_β are matrices with independent $N(0,1)$ Gaussian entries, the work developed by Edelman [13] and Szarek [42] tells us that, with high probability, $\|\Omega_\alpha^{-1}\|_2 \leq c_1\sqrt{k}$ and $\|\Omega_\beta\|_2 \leq c_2 \max(\sqrt{m-k}, \sqrt{k})$. Hence, considering $m - k \geq k$, we get that

$$\sin(\varphi) \leq \left(\frac{\sigma_{k+1}}{\sigma_k}\right)^{2q+1} \|\Omega_\beta \Omega_\alpha^{-1}\|_2 \leq C_\Omega \sqrt{k(m-k)} \left(\frac{\sigma_{k+1}}{\sigma_k}\right)^{2q+1}, \tag{45}$$

where $C_\Omega > 0$ is a constant, holds with high probability. This shows that the angle converges to zero with an exponential rate up to a small rational factor on m and k. Other bounds can be obtained by using another kind of random matrices such as the centered sub-Gaussian random matrices [39] and the Wigner random matrices [43]. Refer to [34] for a recent survey on the different types of random matrices and their spectral properties.

3 Affine Low-rank Approximation

The main objective of this section is to present a low rank approximation of A, which has the form

$$\xi_k := \Big(\sum_{j=1}^{k-1} q_j q_j^T\Big) A(I - zz^T) + (Az)z^T, \quad \forall k = 1, \cdots, \mathrm{rank}(A), \tag{46}$$

where $q_j \in \mathbb{R}^m$ and $z \in \mathbb{R}^n$ are unitary vectors, i.e. multiplying A by two orthogonal projectors on the left and the right and adding a translation matrix. With this aim, we first review a general framework to construct low rank approximations by projecting the columns and rows of A.

Next, in order to select the appropriate vectors q_j and z (and hence the projections), we present a geometric analysis of Householder reflections, studying the optimal choice of the reflector for a general rank-one approximation constructed via a pivoted QR approximation. This analysis sheds light on the construction of approximations over affine subspaces, which we refer to as *affine approximation*. Then, we show that an affine approximation can be written as (46). And later, in Section 5 we numerically show the benefits of this approach.

3.1 Low-Rank Approximation as Projection of Rows and Columns

Consider the matrix $A \in \mathbb{R}^{m\times n}$, with $\mathrm{rank}(A) > k$, and let $\|\cdot\|$ be any unitarily invariant norm. Then, let us construct a low rank approximation using a truncated QR factorization as in equation (11), this is

$$A \approx \bar{Q}\bar{R} = \sum_{j=1}^{k} q_j r_j^T =: \xi_k, \tag{47}$$

where $\bar{Q} \in \mathbb{R}^{m\times k}$ and $\bar{R} \in \mathbb{R}^{k\times n}$, and q_j and r_j are the j-th columns of $\bar{Q}$ and $\bar{R}^T$ respectively. Note that this approximation can also be written as

$$\xi_k = \bar{Q}\bar{Q}^T A = \big(\sum_{j=1}^{k} q_j q_j^T\big)A, \tag{48}$$

and hence, the approximation error is given by

$$\|A - \xi_k\| = \|\big(I - \sum_{j=1}^{k} q_j q_j^T\big)A\| = \|\prod_{j=1}^{k} \big(\underbrace{I - q_j q_j^T}_{=:\mathcal{P}_j}\big)A\|, \tag{49}$$

where the last equality can be easily proved by induction. Hence, the approximation error can be seen as the norm of the matrix obtained after applying k orthogonal projections, $\mathcal{P}_j$, to the columns of A.

In general, we can consider the orthogonal matrices $W = [w_1, \cdots, w_k] \in \mathbb{R}^{m\times k}$ and $Z = [z_1, \cdots, z_k] \in \mathbb{R}^{n\times k}$, and use the orthogonal projectors WW^T and ZZ^T to construct

$$\bar{\xi}_k := WW^T A = \big(\sum_{j=1}^{k} w_j w_j^T\big)A, \quad \text{and} \quad \tilde{\xi}_k := AZZ^T = A\big(\sum_{j=1}^{k} z_j z_j^T\big), \tag{50}$$

for which,

$$\|\bar{\xi}_k - A\| = \|\prod_{j=1}^{k} \big(I - w_j w_j^T\big)A\| \tag{51}$$

$$\|\tilde{\xi}_k - A\| = \|A\prod_{j=1}^{k} \big(I - z_j z_j^T\big)\| = \|\prod_{j=1}^{k} \big(I - z_j z_j^T\big)A^T\|. \tag{52}$$

Then, the approximation errors (51) and (52) are, respectively, the norm of the matrices obtained after applying k orthogonal projections on the columns and rows of A. According to Theorem 2.4, if $w_j = u_j(A)$ or $z_j = v_j(A)$, for $j = 1, \cdots, k$, then the errors (51) and (52) are minimized.

Next, we present the main point of this section, which consist in constructing an approximation by mixing projections of rows and columns, this is

$$\xi_{\bar{r}} := \big(\sum_{j=1}^{s} w_j w_j^T\big)A\big(\sum_{j=1}^{t} z_j z_j^T\big), \tag{53}$$

where $\xi_{\bar{r}}$ is an approximation of A, having at most rank $\bar{r} = \min(s, t, \text{rank}(A))$.

Finally, Lemma 3.1 shows some useful inequalities involving the matrix obtained after projecting the columns of A. Note that it still holds when considering projection of rows instead, by simply applying the same arguments on $Y^T = (I - ZZ^T)A^T$.

Lemma 3.1. Consider $A \in \mathbb{R}^{m\times n}$ and an orthogonal matrix $Z \in \mathbb{R}^{n\times t}$, with $t < \min(m, n)$. Define the matrix $Y = A(I - ZZ^T)$, constructed by orthogonally projecting the columns of A. Then,

$$\sigma_{k+t}(Y) \le \sigma_{k+t}(A) \le \sigma_k(Y). \tag{54}$$

Proof. The left inequality is verified by applying Theorem 2.6 on the product $A(I - ZZ^T)$ with $i = k+t$ and $j = 1$, since an orthogonal projection has unitary norm. To prove the right inequality, define $F := AZ \in \mathbb{R}^{m\times t}$, so that $Y = A - FZ^T$. Next, let Y_{k-1} be the rank $k-1$ truncated SVD approximation of Y, hence

$$\sigma_k(Y) = \|Y - Y_{k-1}\|_2 = \|A - (Y_{k-1} + FZ^T)\|_2 \geq \sigma_{k+t}(A), \tag{55}$$

the last inequality holds since $Y_{k-1} + FZ^T$ is a matrix of rank at most $k + t - 1$. □ □

Corollary 3.2. If $t = 1$, i.e. $Y = A(I - zz^T)$, where $z \in \mathbb{R}^n$ is a unit vector, then

$$\sigma_{k+1}(Y) \leq \sigma_{k+1}(A) \leq \sigma_k(Y), \tag{56}$$

$$\text{rank}(A) - 1 \leq \text{rank}(Y) \leq \text{rank}(A). \tag{57}$$

3.2 Geometric Analysis of a Householder Reflection

The objective of this section is to search a Householder reflector via an optimization problem posed on the set of columns of $A \in \mathbb{R}^{m\times n}$. For this aim, consider $A = [a_1, a_2, \cdots, a_n]$, and let $u \in \mathbb{R}^m$ be any unitary vector and H its corresponding Householder reflector as defined in (9). Next, apply the reflector on the columns of A, this is expressed by the matrix product

$$HA := [h_{a_1}, h_{a_2}, \cdots, h_{a_n}]. \tag{58}$$

Defining $h_u := Hu = \|u\|_2 e_1$, and since a reflection preserves the inner product, we obtain

$$u^T a_j = (h_u)^T h_{a_j} = \|u\|_2 {e_1}^T h_{a_j}, \qquad \text{where } e_1 := (1, 0, \cdots, 0)^T \in \mathbb{R}^m,$$

which indicates that the first component of h_{a_j} is the length of the projection of a_j on u, given as

$$p_j = (u^T a_j)u = \|a_j\|_2 \cos(\varphi_j)u, \tag{59}$$

where $\varphi_j = \angle(u, a_j/\|a_j\|_2)$. Figure 1 shows the reflection across the plane $\mathcal{H}_u$ (defined algebraically by H) of the j-th column of A.

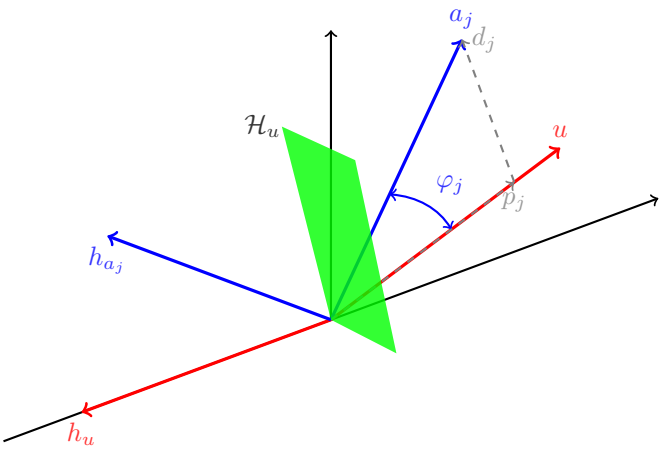


Figure 1: Householder reflection of the vector a_j across the plane $\mathcal{H}_u$. The vectors p_j and d_j denote the projection vectors along and orthogonal to u respectively.

Hence, the product HA can be rewritten as

$$HA = \begin{bmatrix} \|a_1\|_2 \cos(\varphi_1) & \|a_2\|_2 \cos(\varphi_2) & \cdots & \|a_n\|_2 \cos(\varphi_n) \\ r_1 & r_2 & \cdots & r_n \end{bmatrix}, \tag{60}$$

where $r_j \in \mathbb{R}^{m-1}$. Next, note that we can write $H^T = [u, W]$, where $W \in \mathbb{R}^{m\times(m-1)}$ is an orthogonal matrix, then

$$A = H^T HA = [u, W] \begin{bmatrix} \|a_1\|_2 \cos(\varphi_1) & \|a_2\|_2 \cos(\varphi_2) & \cdots & \|a_n\|_2 \cos(\varphi_n) \\ r_1 & r_2 & \cdots & r_n \end{bmatrix}. \tag{61}$$

This implies that the rank-one matrix

$$A_u := u(\|a_1\|_2 \cos(\varphi_1), \cdots, \|a_n\|_2 \cos(\varphi_n)) \tag{62}$$

approximates A with an error, depending on u, given by the functional

$$E(u) = \|A - A_u\|_F = \|W[r_1, \cdots, r_n]\|_F = \|[r_1, \cdots, r_n]\|_F. \tag{63}$$

Next, using the Pythagorean theorem,

$$\|a_j\|_2^2 = \|r_j\|_2^2 + (\|a_j\|_2 \cos(\varphi_j))^2,$$

and hence,

$$\|r_j\|_2^2 = \|a_j\|_2^2(1 - \cos^2(\varphi_j)) = \|a_j\|_2^2 \sin^2(\varphi_j).$$

Then, the functional expressing the approximation error in Frobenius norm is given as

$$E^2(u) = \|A - A_u\|_F^2 = \sum_{j=1}^{n} \|r_j\|_2^2 = \sum_{j=1}^{n} \|a_j\|_2^2 \sin^2(\varphi_j), \tag{64}$$

where each φ_j depends on u.

Setting the optimization problem

From the previous geometric analysis, we note that the approximation error (64) is given as the sum of the squared length of the orthogonal projections

$$d_j = \|a_j\|_2 \sin(\varphi_j) u_{\perp j}, \tag{65}$$

where $u_{\perp j}$ is a unit vector orthogonal to u, see Figure 1. Then, the error functional takes the form

$$E^2(u) = \sum_{j=1}^{n} \|d_j\|_2^2 = \sum_{j=1}^{n} \|a_j\|_2^2 \sin^2(\varphi_j). \tag{66}$$

Then, the minimum of E is also a solution of the optimization problem consisting in finding a line, passing through the origin in the m dimensional space, such that the sum of squared orthogonal distances from the points a_j's to it is minimized. This problem is well known in statistics and corresponds to the solution of the *total least-square problem*, cf. [30].

Definition 3.3. For a given matrix $A \in \mathbb{R}^{m\times n}$, we define its best fitting line as

$$\mathcal{L}_A(\tau) = \tau u, \quad \tau \in \mathbb{R}, \tag{67}$$

where u is the minimizer of $E(u)$ defined in (66).

By Theorem 2.4, see also [30, Thm. 5], the truncated SVD provides a best fitting line by setting $u = u_1(A)$, and this solution is unique whenever $\sigma_1 \neq \sigma_2$. This is, we have $A_u = u_1(A)\sigma_1(A)v_1(A)^T$.

Next, if we do not consider the restriction that the best fitting line passes through the origin, then we obtain a general best fitting line (since we remove the restriction of passing through the origin), see the calculus in [40, Appendix A.7] (we also present an analysis in appendix A.1). For this case, the best fitting line is

$$\mathcal{L}_g(\tau) = g + \tau u_1(Y), \quad \tau \in \mathbb{R}, \tag{68}$$

where ,

$$g := \frac{1}{n}\sum_{j=1}^{n} a_j, \tag{69}$$

$$Y := [a_1 - g, \cdots, a_n - g] = A - gc^T, \tag{70}$$

where g is known as the gravity center of A, $c = (1, \cdots, 1)^T \in \mathbb{R}^n$, and Y can be regarded as the matrix obtained by centering the columns of A with respect to g.

Figure 2 shows graphically both lines $\mathcal{L}_A$ and $\mathcal{L}_g$ for a matrix whose columns are points of $\mathbb{R}^3$.

Next, observe that the total least-squares misfit is smaller for the line $\mathcal{L}_g$, since it is the solution of the non-restricted optimization problem. In matrix terminology, it means that

$$\|A - gc^T - Y_k\|_F \leq \|A - A_k\|_F, \tag{71}$$

where Y_k and A_k are the rank-k truncated SVD approximations of Y and A respectively, as defined in (5).

Finally, note that $Y = A - gc^T$ can be rewritten as

$$Y = A\underbrace{(I - \frac{1}{n}cc^T)}_{\mathcal{P}}, \tag{72}$$

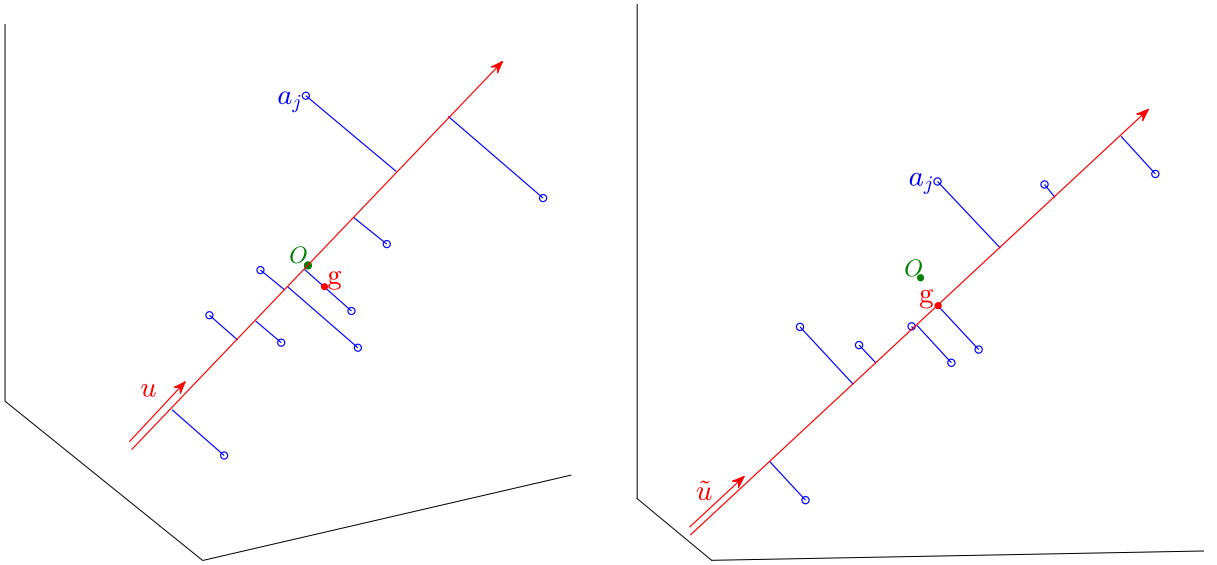
where $\mathcal{P}$ is a rank $n-1$ orthogonal projector, and hence, the results from Corollary 3.2 hold for Y.

3.3 Getting an Affine Low-Rank Approximation

Below, we express the numerical analysis made in the previous subsection by means of Algorithm 3.1, which shows the procedure to construct an affine approximation for any real matrix. This is, approximate the matrix $A \in \mathbb{R}^{m\times n}$ as

$$A \approx gc^T + \xi_{k-1}, \tag{73}$$

where ξ_k is a rank-$(k-1)$ approximation of $Y = A(I - \frac{1}{n}cc^T)$.



(a) Line $\mathcal{L}_A(\tau) = \{\tau u \in \mathbb{R}^3 |\ \tau \in \mathbb{R}\}$ passing through the origin, $u = u_1(A)$.

(b) Line $\mathcal{L}_g(\tau) = \{g + \tau\tilde{u} \in \mathbb{R}^3 |\ \tau \in \mathbb{R}\}$ passing through the gravity center of A, $\tilde{u} = u_1(Y)$.

Figure 2: Best fitting lines (represented as arrows) of a matrix $A = [a_1, \cdots, a_n] \in \mathbb{R}^{3\times n}$. The small circles represent the columns a_j's, for $j = 1, \cdots, n$, and their projections over the lines are also showed. The gravity center g and the matrix Y are defined in (69) and (70) respectively.

Algorithm 3.1 $[\xi_k]$ = ALORA(A,k)

Require: $A \in \mathbb{R}^{m\times n}$.
Returns: A rank-k approximation of A.
1: $c = (1, \cdots, 1)^T \in \mathbb{R}^n$.
2: $Y = A(I - \frac{1}{n}cc^T)$.
3: Compute ξ_{k-1}, a rank-$(k-1)$ approximation of Y.
4: $\xi_k = gc^T + \xi_{k-1}$.
5: **return** ξ_k.

We name Algorithm 3.1 *ALORA* (short for Affine Low-Rank Approximation). Its computational complexity is $\mathcal{O}(mnk)$ with a constant factor depending on the use of QRCP or subspace iteration in line 2 of the algorithm.

Error analysis

Below we present how to easily derive a bound for the approximation error of an affine approximation. First, consider that

$$\|Y - \xi_{k-1}\|_2 \leq \mathcal{F}_Y \sigma_k(Y), \tag{74}$$

where $\mathcal{F}_Y$ is a function depending on the low rank method used. Then, since $\xi_k = gc^T + \xi_{k-1}$, we obtain

$$\|A - \xi_k\|_2 = \|Y - \xi_{k-1}\|_2 \leq \mathcal{F}_Y \sigma_k(A), \tag{75}$$

where we use the fact that $\sigma_k(Y) \leq \sigma_k(A)$ ensured by Corollary 3.2.

Next, as explained in Section 3.2, the approximation ξ_k can be interpreted as fitting the columns of the matrix into an affine subspace of dimension $k-1$. And since the rank-k truncated SVD can be seen as fitting into a subspace of dimension k, we might also use an affine subspace of dimension k. In terms of matrices, it means that Y is approximated by a rank-k matrix ξ_k, and the affine approximation for A is constructed as

$$A \approx \xi_{k+1} := gc^T + \xi_k \quad \text{with} \quad \|A - \xi_{k+1}\|_2 = \|Y - \xi_k\|_2 \leq \mathcal{F}_Y \sigma_{k+1}(A), \tag{76}$$

where it should be noted that the rank of ξ_{k+1} is bounded by $k+1$.

In Section 5 we plot the approximation errors when A is approximated by ξ_k and ξ_{k+1}, showing that in many cases they both overcome the QRCP approximation of rank-k. The numerical experiences also show that bounds (75) and (76) are pessimistic practice.

Finally, note that a derivation of a bound for the approximation error can be obtained for any low-rank approximation method by using the fact that $\|A - \xi_{k+1}\|_2 = \|Y - \xi_k\|_2$. And depending on the method, we can obtain a bound depending on $\sigma_{k+1}(A)$. For instance, for a QR based approximation, simply by replacing $\mathcal{F}_Y$ by its appropriate value, see (16). And for subspace iteration, simply by using the fact that $\sigma_{k+1}(Y) \leq \sigma_{k+1}(A)$ when bounding $\|Y - \xi_k\|_2$ using (30).

4 Correlation of Matrices Using their Gravity Center

In the previous section we have shown how to construct an affine low rank approximation for any matrix $A \in \mathbb{R}^{m\times n}$. In this section we explore the structural relation of a matrix and their best fitting lines $\mathcal{L}_A$ and $\mathcal{L}_g$ studied in the previous section, this allows us to understand for which kind of matrices an affine low rank approximation could be better than the non-affine one. We define a correlation coefficient that helps to understand the matrix structure seeing its columns as spatial points in $\mathbb{R}^m$. We start by analyzing a particular case of matrices with exponentially decreasing singular values. The analysis from this section leads to the construction of an algorithm that provides a low-rank approximation that performs very well, in particular for matrices having singular decreasing singular values, moreover it also provides an estimate of the matrix norm.

4.1 Matrices with Exponentially Decreasing Singular Values

In many important problems of linear algebra oriented to mathematical modeling, matrix compression and related subjects, we handle a matrix $A \in \mathbb{R}^{m\times n}$ with singular values that decrease exponentially, this means that if A has singular triplets (u_j, v_j, σ_j) for $j = 1, \cdots, r = \text{rank}(A)$, then

$$\sigma_j \leq q^j \sigma_1, \tag{77}$$

where $0 < q < 1$. Such matrix arises, for example, as an "admissible" block in the context of discretization of boundary integral operators [2]. They are also interesting in merely theoretical and testing problems such as the Kahan matrix [8]. In order to see if an affine low-rank approximation would be useful for these kind of matrices, let us write the gravity center of A using its singular triplets,

$$g = \frac{1}{n}\sum_{j=1}^{n} a_j = \frac{1}{n}\sum_{l=1}^{r} u_l \sigma_l \left(\sum_{j=1}^{n} v_l(j)\right), \tag{78}$$

then,

$$g = \sigma_1 \left(\sum_{l=1}^{r} \tilde{\nu}_l u_l \right), \quad \text{with} \quad \tilde{\nu}_l = \frac{\sigma_l}{n\sigma_1} \left(\sum_{j=1}^{n} v_l(j) \right), \tag{79}$$

and since $\sum_{j=1}^{n} |v_l(j)| \leq \sqrt{n}$, then $|\tilde{\nu}_l| \leq \frac{q^l}{\sqrt{n}}$. Hence, if the singular values of A decrease as in equation (77), then the unitary vector in the direction of g would be a good approximation u_1, and this approximation gets better when q gets smaller. In other words, the matrix A is such that its best fitting lines, $\mathcal{L}_A$ and $\mathcal{L}_g$, almost overlap.

Note, that applying Algorithm 3.1 to a matrix with rapidly singular values can produce an increase on the precision as in the case of Figure 3, and for some cases as in Figures 6 and 7 it may not produce good results. However, in all the cases of matrices with exponentially decreasing singular values, we will get interesting characterizations of their singular triplets, as it is shown in the next subsection.

Finally, a useful observation, to which we will refer later, is that if A has exponentially decreasing singular values, then the cosine of the angle made by the gravity center and its j-th column is closer to 1 when q gets small, this is true since

$$\frac{g^T a}{\|g\|_2 \|a_j\|_2} = \frac{\tilde{\nu}_1 v_1(j) + \sum_{l=2}^{r} \left(\frac{\sigma_l}{\sigma_1}\right)^2 \tilde{\nu}_l v_l(j)}{\sqrt{\tilde{\nu}_1^2 + \sum_{l=2}^{r} \left(\frac{\sigma_l}{\sigma_1}\right)^2 \tilde{\nu}_l^2} \sqrt{v_1(j)^2 + \sum_{l=2}^{r} \left(\frac{\sigma_l}{\sigma_1}\right)^2 v_l(j)^2}}. \tag{80}$$

4.2 Characterization of Matrices using their Gravity Center

Consider the matrix $A \in \mathbb{R}^{m \times n}$, from the previous best fitting line analysis, it is clear that a sufficient condition for the lines $\mathcal{L}_A$ and $\mathcal{L}_g$ to coincide, is that $g = 0$. Let us consider the reverse case, i.e. if $\mathcal{L}_A$ and $\mathcal{L}_g$ are identical, then what can we say about the matrix A?. The following theorem provides the answer.

Theorem 4.1. *Consider $A \in \mathbb{R}^{m \times n}$, with $r = \operatorname{rank}(A)$ and singular triplets (u_j, v_j, σ_j), for $j = 1, \cdots, r$. Let its best fitting lines be $\mathcal{L}_A$ and $\mathcal{L}_g$ as defined in (67) and (68) respectively. Consider the vector of ones $c = (1, \cdots, 1)^T \in \mathbb{R}^n$. Then, both lines are identical if and only if*

$$A = B + \|g\|_2 u_1 c^T, \tag{81}$$

where $B \in \mathbb{R}^{m \times n}$ is a matrix for which the gravity center of its columns is zero. Furthermore, if $\mathcal{L}_A$ and $\mathcal{L}_g$ are identical, then the norm of A is bounded as

$$\|A\|_2 \geq \sqrt{n} \|g\|_2, \tag{82}$$

and if $\|g\|_2 \neq 0$, we get

$$u_1 = \frac{g}{\|g\|_2}, \tag{83}$$

and the right singular vectors hold

$$v_1^T c = \sum_{i=1}^{n} v_1(i) = \frac{n\|g\|_2}{\sigma_1}, \tag{84}$$

$$v_j^T c = \sum_{i=1}^{n} v_j(i) = 0, \quad \text{for } j = 1, \cdots, r. \tag{85}$$

Proof. If $g = 0$, the first statement follows straightforwardly. Hence, let us consider the non-trivial case when $g \neq 0$. If $A = B + \|g\|_2 u_1 c^T$, then clearly both lines coincide, since for this case when computing g we obtain

$$u_1 = \frac{g}{\|g\|_2}, \tag{86}$$

where we use the fact that the gravity center of B is zero. To prove the reverse statement, assume both lines are identical, i.e. assume that (86) holds. Then, define $B := A - gc^T$, where clearly the gravity center of B is zero. And using (86) we can write

$$A = B + \|g\|_2 u_1 c^T, \tag{87}$$

which proves the first statement of the theorem.

Next, to prove the second statement of the theorem, write the j-th column of A using its singular triplets, this is

$$a_j = \sum_{l=1}^{r} u_l \sigma_l v_l(j), \quad \text{for } j = 1, \cdots, n. \tag{88}$$

By definition of the gravity center and (86), we get

$$g = \frac{1}{n} \sum_{j=1}^{n} a_j = \|g\|_2 u_1, \tag{89}$$

and combining (88) and (89), we get

$$\sum_{l=1}^{r} u_l \sigma_l \left(\sum_{j=1}^{n} v_l(j) \right) = n \|g\|_2 u_1, \tag{90}$$

$$\underbrace{\left(\sigma_1 \left(\sum_{j=1}^{n} v_1(j) \right) - n\|g\|_2 \right)}_{\beta_1} u_1 + \sum_{l=2}^{r} \underbrace{\left(\sigma_l \sum_{j=1}^{n} v_l(j) \right)}_{\beta_l} u_l = 0, \tag{91}$$

and since (91) is a linear combination of linearly independent vectors, then $\beta_1 = \beta_2 = \cdots = \beta_r = 0$, which proves (84) and (85). Finally, by the Cauchy-Schwartz inequality, we have that

$$|v_1^T c| \leq \|c\|_2 = \sqrt{n}, \tag{92}$$

and (82) follows by replacing (84) on (92). □ □

Next, let us explore a direct consequence of the previous theorem. First, note that

$$\frac{\sqrt{n}}{\sqrt{r}} \|a_m\|_2 \leq \|A\|_2 \leq \sqrt{n} \|a_M\|_2, \tag{93}$$

where a_m and a_M are, respectively, the columns of A with smallest and largest norm. These inequalities follow from the fact that $\frac{1}{\sqrt{r}}\|A\|_F \leq \|A\|_2 \leq \|A\|_F$.

Hence, when A is such that its best fitting lines, $\mathcal{L}_A$ and $\mathcal{L}_g$, are identical, then we can obtain a narrow bound for the matrix norm, given as

$$\sqrt{n}\|g\|_2 \leq \|A\|_2 \leq \sqrt{n}\|a_s\|_2, \tag{94}$$

and we can obtain an estimate of the norm that becomes more precise when the columns of the matrix have similar norm. However, it is not evident when A is such that $\mathcal{L}_A$ and $\mathcal{L}_g$ are identical, we explore this in the next subsection.

Finally, gathering the results from this and the previous subsections, we get that an affine approximation should not be used when the gravity center of the columns of the matrix is very small, since for this case both best fitting line coincide, e.g. the matrix $A = \texttt{randn}(n)$ constructed with `MATLAB`, has as entries normally distributed random numbers having mean zero, so an affine approximation would not make sense. For all other cases, an affine approximation might increase the precision as it is shown in Section 5.

4.3 Measuring the Correlation of Matrices

We can obtain insights about the geometrical distribution of the columns of a matrix by using formal concepts from statistics, as the correlation of a matrix.

The correlation of a matrix $A \in \mathbb{R}^{m\times n}$ is typically expressed using the pairwise correlation of its columns, this is, consider the columns a_j and a_l with means $\bar{g}_j := \frac{1}{m}\sum_{i=1}^{m} a_j(i)$ and $\bar{g}_l := \frac{1}{m}\sum_{i=1}^{m} a_l(i)$ respectively. Then, we can obtain the *Pearson correlation* coefficient defined as

$$\rho_{jl} = \frac{\sum_{i=1}^{n}(a_j(i)-\bar{g}_j)(a_l(i)-\bar{g}_l)}{\sqrt{\sum_{i=1}^{n}(a_j(i)-\bar{g}_j)^2}\sqrt{\sum_{i=1}^{n}(a_l(i)-\bar{g}_l)^2}} = \frac{\bar{a}_j^T}{\|\bar{a}_j\|_2}\frac{\bar{a}_l}{\|\bar{a}_l\|_2} = \cos(\angle(\bar{a}_j,\bar{a}_l)), \tag{95}$$

where $\bar{a}_j := a_j - \bar{g}_j\bar{c}$ and $\bar{a}_l = a_l - \bar{g}_l\bar{c}$ are obtained by centering a_j and a_l with respect to their mean, with $\bar{c} = (1,\cdots,1)^T \in \mathbb{R}^m$. This provides a symmetric $(m\times n)$ matrix of coefficients ρ_{jl} having ones on the diagonal. For example, the function `corr` from `MATLAB` gives exactly this matrix. And since the pairwise interaction of distinct columns can provide at most $n(n-1)/2$ different values, then we can define the correlation of a matrix as a real number, given as

$$\mathcal{C}(A) := 2\frac{\sum_{i<j}|\rho_{ij}|}{n(n-1)}.$$

Note that $0 \leq \mathcal{C}(A) \leq 1$. It is clear that at a given stage of the approximation, computing $\mathcal{C}(A)$ for all the columns would provide an accurate stopping criterium. For instance, at the step $k-1$ of one approximation algorithm, consider

$$F = A - \xi_{k-1}, \tag{96}$$

then, theoretically if $\mathcal{C}(F) = 1$, then F is a rank-one matrix and the algorithm should stop at the step k. This could be replaced by the weaker condition $1-\delta < \mathcal{C}(F)$. This technique could be used as a stopping criterium, however costly, indeed it would cost $\mathcal{O}(mn^2k)$.

Next, we propose a cheaper way to measure the correlation of A by defining the correlation vector $\tilde{\rho}_A \in \mathbb{R}^n$ as

$$\tilde{\rho}_A(j) := \frac{g^T a_j}{\|g\|_2 \|a_j\|_2}, \tag{97}$$

which cost $\mathcal{O}(mn)$ to compute, and the correlation coefficient

$$\mathcal{G}(A) := \frac{\max(\tilde{\rho}_A) - \min(\tilde{\rho}_A)}{2}. \tag{98}$$

Note that $0 \leq \mathcal{G}(A) \leq 1$ and that $\mathcal{G}(A)$ is a good indicator of the spacial distribution of the columns of the matrix with respect to its gravity center. Furthermore, we can approximate $\mathcal{G}(A)$ with an small cost of $\mathcal{O}(ml)$, $l < n$, when using a randomized approach. For instance, the randomized version of QRCP [11] obtains its permutation by applying the classic QRCP on the smaller matrix $\Omega_r A \in \mathbb{R}^{l\times n}$, where $\Omega_r \in \mathbb{R}^{l\times m}$ is a random compression matrix. Hence, we can use (or reuse) a compression matrix in a randomized algorithm to approximate the gravity center of A by the gravity center of $A\Omega_c$, where $\Omega_c \in \mathbb{R}^{n\times l}$, and make the approximation $\mathcal{G}(A) \approx \tilde{\rho}(A\Omega)$. Moreover, note that the approximation of g holds

$$\|g - \tilde{g}\|_2 = \|\frac{1}{n}Ac - \frac{1}{n}A\Omega c\|_2 \leq \frac{\|A\|_2(1 + \|\Omega\|_2)}{\sqrt{n}}, \tag{99}$$

where $\tilde{g}$ is the gravity center of $A\Omega$, and $c = (1, \cdots, 1)^T \in \mathbb{R}^n$, and this approximation is justified when the norms of A and Ω are small.

4.4 Matrices with High Correlation

We consider that a matrix A has high correlation if the mean of the correlation vector $\tilde{\rho}_A$ defined in (97) is close to 1, or if the correlation coefficient $\mathcal{G}(A)$, defined in (98), is close to 0. In order to find a representation of matrices with high correlation, let us consider a rank-one matrix A, from the linear dependency of its columns it is clear that its correlation coefficients, $\mathcal{C}(A)$ and $\mathcal{G}(A)$, are equal to 1. Furthermore it is clear that A can be written as $A = [\beta_1 \texttt{ones}(m, n_1), \cdots, \beta_k \texttt{ones}(m, n_k)]$, with appropriate coefficients β_j and $n_1 + \cdots + n_k = n$. Next lemma gives us an useful representation of A.

Lemma 4.2. Consider $A = [\beta_1 \texttt{ones}(m, n_1), \cdots, \beta_k \texttt{ones}(m, n_k)] \in \mathbb{R}^{m\times n}$, where $\beta_j \in \mathbb{R}$ for $j = 1, \cdots, k$, and $n_1 + \cdots + n_k = n$. Then,

$$\texttt{abs}(u_1(A)) = \texttt{abs}\left(\frac{g}{\|g\|_2}\right), \quad \texttt{abs}(v_1(A)) = \texttt{abs}\left(\frac{g_t^T}{\|g_t\|_2}\right), \tag{100}$$

and,

$$\sigma_1(A) = \sqrt{m(\sum_{j=1}^{k} n_j \beta_j^2)}, \tag{101}$$

where where g and g_t are the gravity centers of the columns of A and A^T respectively.

Proof. First, note that the line passing through the origin of $\mathbb{R}^m$ in the direction of vector $c_1 = \texttt{ones}(m, 1) \in \mathbb{R}^m$ is the best fitting line of the columns of A, and since clearly g also belongs to this line, it means that both best fitting lines of A, i.e. $\mathcal{L}_A$ and $\mathcal{L}_g$, coincide, and using Theorem 4.1 we get the left equality of (100). And analogously, to obtain the right equality of (100), observe that the line passing through the origin of $\mathbb{R}^n$ in the direction of the vector $c_1 = (\beta_1 \texttt{ones}(1, n_1), \cdots, \beta_k \texttt{ones}(1, n_k))^T \in \mathbb{R}^n$ is the best fitting line of the columns of A^T and it contains g_t, then apply Theorem 4.1 on A^T.

Next, note that A has rank-one, this is $\sigma_j(A) = 0$ for $j \geq 2$, hence both spectral and Frobenius norm coincide and a simple calculus shows that $\sigma_1(A) = \|A\|_F = \sqrt{m(\sum_{j=1}^{k} n_j \beta_j^2)}$. □ □

Next, let us propose a rank-one approximation for matrices having high correlation. Note that in particular, by (80) we know that a matrix A with exponentially decreasing singular values tends to have high correlation. We define the rank-one approximation of these kind of matrices as

$$A \approx \xi_1 := \frac{g}{\|g\|_2} \tilde{\sigma}_1 \frac{g_t^T}{\|g_t\|_2}, \tag{102}$$

where $\tilde{\sigma}_1$ approximates $\sigma_1(A)$, and according to Lemma 4.2 it could be taken as $\tilde{\sigma}_1 := \sqrt{mn} * \texttt{mean}(A(:))$. However, our experiences show that a better approximation of the first singular value is (see the analysis made in Section 4.2)

$$\sigma_1 \approx \tilde{\sigma}_1 := \|g\|_2 \sqrt{n}. \tag{103}$$

In Section 5, we show that $\tilde{\sigma}_1$ approximates very well $\sigma_1 = \|A\|_2$ for most of the test matrices, even though most of them do not have singular values decreasing exponentially.

For our next algorithm, we need the following definition.

Definition 4.3. For a vector $w \in \mathbb{R}^m$ we define its sign as

$$S(v) := \texttt{sign}\left(\sum_{i=1}^{m} \texttt{sign}\big(w(i)\big) \right),$$

where `sign` is the standard function for real numbers.

The following algorithm, named AGC (short for Approximation by Gravity Centers) has an empirical approach to compute a rank-k approximation of a matrix A, it uses (102) as the rank-one approximation and then update iteratively the matrix by a Householder based technique choosing as Householder vector a column that has maximal first component. The complexity AGC is $\mathcal{O}(mnk)$ but with a constant factor much smaller than QRCP or subspace iteration. Using some of the ideas presented in this section, which led to the construction of AGC, currently the authors are working on a linear-cost algorithm for matrices with exponentially decreasing singular values.

Algorithm 4.1 $[A_k] = \text{AGC}(A,k)$

Require: $A = [a_1 \;\; a_2 \;\; \cdots \;\; a_n] \in \mathbb{R}^{m \times n}$.
Returns: Rank-k approximation of A.

$c = (1, \cdots, 1)^T \in \mathbb{R}^n$, $\tilde{c} = (1, \cdots, 1)^T \in \mathbb{R}^m$.
$g = S(g)(1/n)Ac, \quad \tilde{\sigma}_1 = \|g\|_2 \sqrt{n}, \quad g_t = (1/m)A^T \tilde{c}$.
$\xi_k = \frac{g}{\|g\|_2} \tilde{\sigma}_1 \frac{g_t^T}{\|g_t\|_2}$.
$Y = A - \xi_k$.
for $i = 2 \to k$ **do**
  Find j such that $Y(1,j) = \max(Y(1,l) \mid \text{for } l = 1 \cdots n)$.
  $u = \frac{Y(:,j)}{\|Y(:,j)\|_2}$.
  $Y = Y - u(u^T Y)$.
  $\xi_k = \xi_k + u(u^T Y)$.
end for
return ξ_k.

5 Numerical Experiments

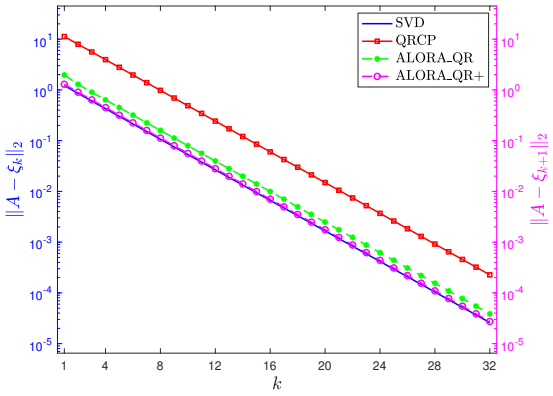
5.1 Low-rank Approximation of Challenging Matrices

In this section we numerically show the benefits of algorithms 3.1 and 4.1 on a set of challenging matrices with $m = n = 256$, given in Table 1. Most of the matrices from Table 1 have been previously used in experiments with QR algorithms [7, 19]. These matrices have been constructed using `MATLAB` and they are easy to replicate for testing and verification. Some of the test matrices, have the form $A = U\Sigma V^T$ where, when it is not specified, U and V are random orthogonal matrices and $\Sigma = \mathrm{diag}(\sigma_1, \cdots, \sigma_n)$ is a diagonal matrix containing prescribed singular values, the machine epsilon is given as $\epsilon = 2.22E-16$.

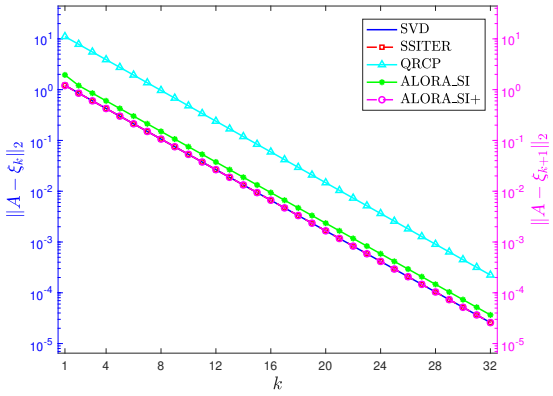
Table 1: Test matrices

No.	Matrix	Description
1	`BAART`	Coming from the discretization of the first kind Fredholm integral equation, cf. [23].
2	`BREAK-1`	$A = U\Sigma V^T$, where Σ is such that $\sigma_1 = \cdots = \sigma_{n-1} = 1$, and $\sigma_n = 10^{-9}$, cf. [3].
3	`BREAK-9`	$A = U\Sigma V^T$, where Σ is such that $\sigma_1 = \cdots = \sigma_{n-9} = 1$, and $\sigma_{n-8} = \cdots = \sigma_n = 10^{-9}$, cf. [3].
4	`DERIV2`	Coming from the computation of the second derivative, cf. [23].
5	`EXPON`	$A = U\Sigma V^T$, where Σ is such that $\sigma_1 = 1$, and for $i = 2, \cdots, n$ the singular values are $\sigma_i = \alpha^{i-1}$, cf. [3].
6	`FOXGOOD`	Coming from the discretization of the first kind Fredholm integral equation of a severely ill-posed problem, first used by Fox and Goodwin, cf. [23].
7	`GKS`	Upper-triangular matrix whose j-th diagonal element is $1/\sqrt{j}$ and whose (i,j) element is $-1/\sqrt{j}$ for $j > i$, cf. [21, 16].
8	`GRAVITY`	Coming from the discretization of a one-dimensional model problem in gravity surveying, cf. [23].
9	`HC`	$A = U\Sigma V^T$, where Σ has diagonal entries 100, 10, and the following $n-2$ are evenly spaced between 10^{-2} and 10^{-8}, cf. [26].
10	`HEAT`	Inverse heat equation, cf. [23].
11	`PHILLIPS`	Phillips test problem, cf. [23].
12	`RANDOM`	Random matrix $A = 2 * \mathtt{rand}(n) - 1$, cf. [21].
13	`SCALE`	A random matrix whose i-th row is scaled by the factor $\eta^{i/n}$, with $\eta = 10\epsilon$, cf. [21].
14	`SHAW`	1D image restoration model, cf. [23].
15	`SPIKES`	Test problem with a "spiky" solution, cf. [23].
16	`STEWART`	Matrix $A = U\Sigma V^T + 0.1\sigma_n * \mathtt{rand}(n)$, where Σ has first half of the diagonals decreasing geometrically from 1 to $\sigma_n = 10^{-3}$, and the last half of the diagonals being set to zero, cf. [41].
17	`URSELL`	Coming from the discretization of an integral equation with no square integrable solution, cf. [23].
18	`WING`	Coming from a test problem with a discontinuous solution, cf. [23].
19	`KAHAN`	The Kahan matrix, cf. [27].
20	`DEVIL`	Devil stairs matrix, a matrix with gaps in its singular values, cf. [41].
21	`RAND-UNIF`	Random matrix with uniformly distributed entries, $A = \mathtt{rand}(n)$.
22	`3D-LAP-ADM`	Laplacian kernel evaluated on a 3D admissible domain, see description in Section 5.4.
23	`3D-LAP-NADM`	Laplacian kernel evaluated on a 3D non-admissible domain, see description in Section 5.4.

Next, we present the approximation error for a rank-k approximation of different test matrices, for $k = 1, \cdots, 32$. We compare ALORA with QRCP and subspace iteration. Figures 3 to 7 show the approximation errors for some of the test matrices, in order to appreciate the cases where an affine low rank approximation is advantageous or disadvantageous. The labels ALORA_QR and ALORA_SI refer to ALORA using QRCP and subspace iteration (using just small parameters $q = 1$ and $l = k + 3$) to produce the rank $k-1$ approximation needed in line 3 of Algorithm 3.1. All figures include a right Y-axis where the values ALORA_QR+ and ALORA_SI+ are plotted, they are obtained by plotting for a given k, the error made by approximating A by the matrix ξ_{k+1} defined in (76). Note that the curves of the SVD, SSITER and ALORA_SI+ almost overlap each other.

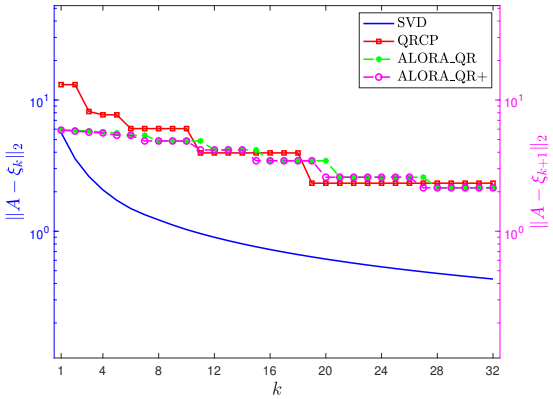


(a) Comparison of the approximation error of ALORA, created with QRCP, with respect to standard methods.

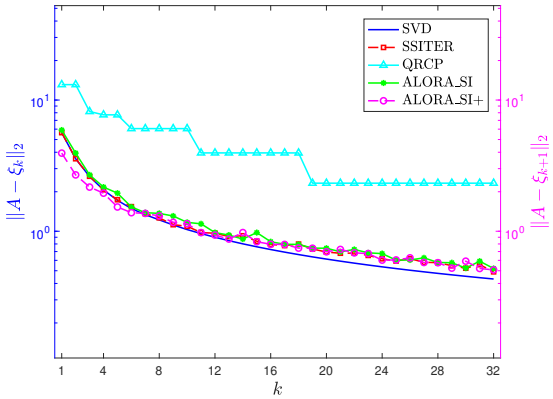


(b) Comparison of the approximation error of ALORA, created with subspace iteration, with respect to standard methods.

Figure 3: Convergence curves of the approximation error for the KAHAN matrix.



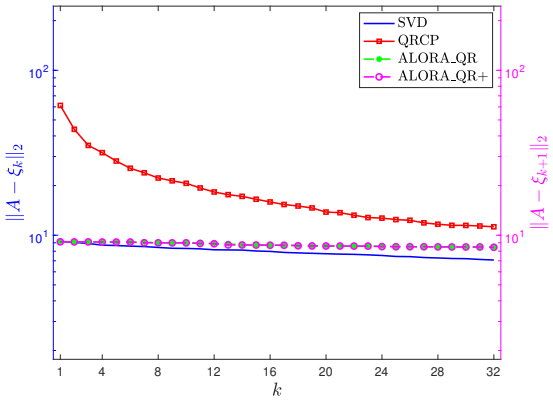
(a) Comparison of the approximation error of ALORA, created with QRCP, with respect to standard methods.



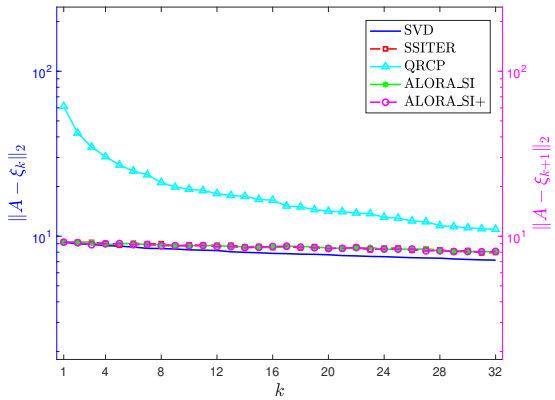
(b) Comparison of the approximation error of ALORA, created with subspace iteration, with respect to standard methods.

Figure 4: Convergence curves of the approximation error for the GKS matrix.

Note that for the matrices with slowly decreasing singular values, GKS and RAND-UNIF, we have that ALORA improves the approximation for k small. While for the other cases, when the matrices have rapidly decreasing singular values, as studied in Section 4.1, their best fitting lines tend to overlap each other and hence an affine approximation may increase considerably the precision as in the case of

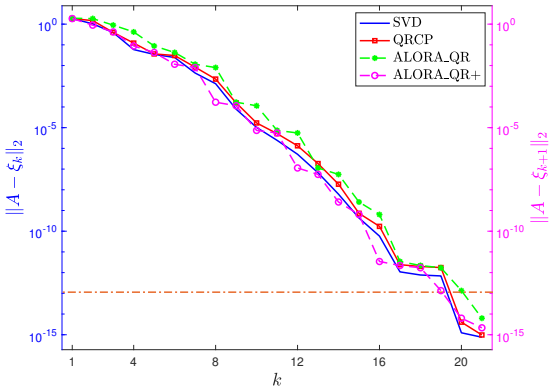


(a) Comparison of the approximation error of ALORA, created with QRCP, with respect to standard methods.

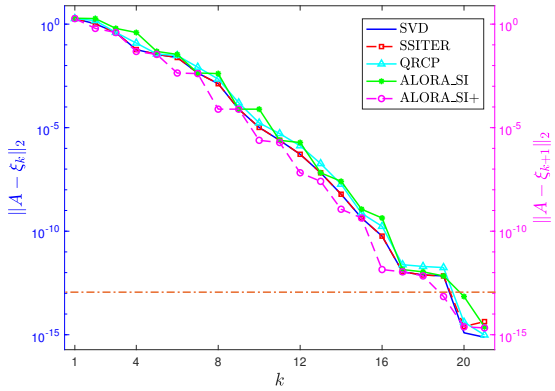


(b) Comparison of the approximation error of ALORA, created with subspace iteration, with respect to standard methods.

Figure 5: Convergence curves of the approximation error for the RAND-UNIF matrix.



(a) Comparison of the approximation error of ALORA, created with QRCP, with respect to standard methods.

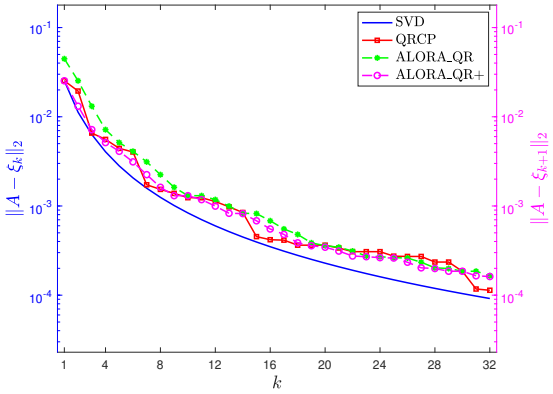


(b) Comparison of the approximation error of ALORA, created with subspace iteration, with respect to standard methods.

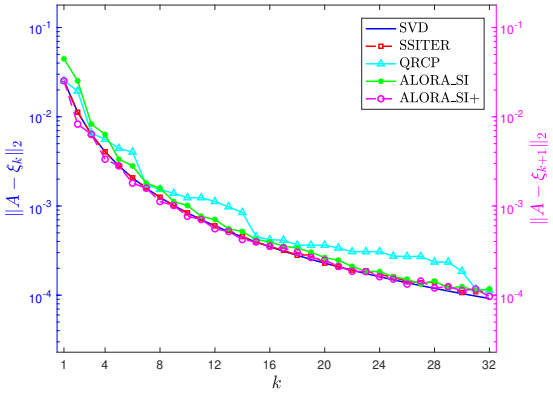
Figure 6: Convergence curves of the approximation error for the SHAW matrix. The horizontal line represents the threshold value, $\epsilon \max(m, n)\|A\|_2$, beyond which the singular values are considered as zero.

Figure 3, and for some cases as in Figures 6 and 7 it may not produce good results since the rank-one approximation gc^T, used by the ALORA algorithm, might be far from the optimal, for these cases the Algorithm AGC might produce better results, since it starts with a better rank-one approximation.

Next, we compute the approximation errors for all the matrices described in Table 1. Considering an approximation of rank $k = 1, \cdots, \min(\text{rank}(A), 16)$, we compute the errors



(a) Comparison of the approximation error of ALORA, created with QRCP, with respect to standard methods.



(b) Comparison of the approximation error of ALORA, created with subspace iteration, with respect to standard methods.

Figure 7: Convergence curves of the approximation error for the DERIV2 matrix.

$$E_{QRCP}(k) = \frac{\|A - \xi_k\|_2}{\sigma_{k+1}(A)}, \tag{104}$$

$$E_{ALORA\_QR+}(k) = \frac{\|A - gc^T - \bar{\xi}_k\|_2}{\sigma_{k+1}(A)}, \tag{105}$$

$$E_{ALORA\_SI+}(k) = \frac{\|A - gc^T - \tilde{\xi}_k\|_2}{\sigma_{k+1}(A)}, \tag{106}$$

where ξ_k and $\bar{\xi}_k$ are rank-k approximations of A and Y respectively constructed using QRCP, and $\tilde{\xi}_k$ is a rank-k approximation of Y constructed using subspace iteration (Algorithm 2.1). Figure 8 plots the average of these values for all the matrices from Table 1.

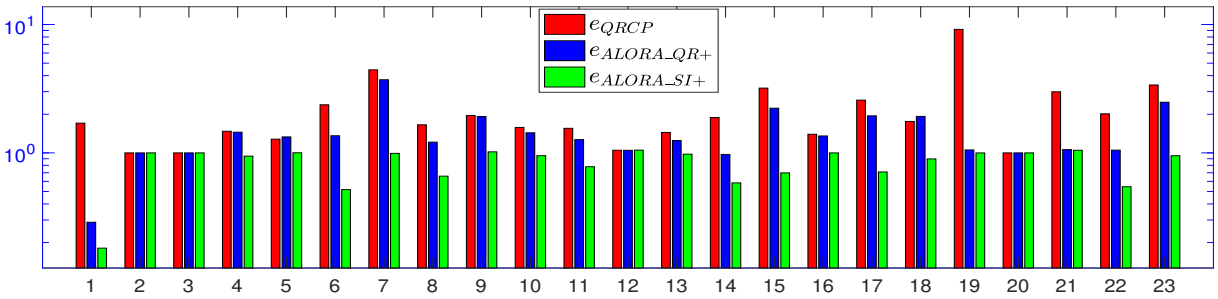


Figure 8: Mean of the ratios of the errors of rank-k approximations created by ALORA_QR+ and ALORA_SI+ to the optimal error. For each matrix, e_{QRCP}, e_{ALORA_QR+} and e_{ALORA_SI+} are, respectively, the mean of the vectors E_{QRCP}, E_{ALORA_QR+} and E_{ALORA_SI+} defined in (104), (105) and (106).

We can clearly see the improvement of using ALORA in its both versions, ALORA_QR+ and ALORA_SI+. Note that the former performs, in average, better than QRCP, while the latter overpasses the accuracy of the other methods. Hence, constructing the rank-k approximation of a matrix as fitting its columns into a k-dimensional affine subspace can improve the accuracy of the approximation.

5.2 Approximation of the Matrix Norm

Using the analysis done in Section 4, we show that our estimate $\tilde{\sigma}_1 = \|g\|_2\sqrt{n}$, given in (103), for the norm of a given matrix A, works quite good for most of the test matrices from Table 1. We compare this estimate with the one obtained performing a truncated QRCP Factorization of A, i.e. $A = QRP^T$ as in (11), where generally authors approximates the i-th singular value as $|R_{i,i}|$, see e.g. [7, Sec. 4], [19, 41]. However, this estimate is rough and more precisely viewing QRCP as the decomposition of type (47), an estimate of σ_i can also be taken as $\|R(i,:)\|_2$. Note that there are more precise ways to approximate the norm using a QR based method, for example we can use the L-values (or the more strong algorithms) proposed by Stewart [41, Sec. 6].

In Figure 9 we plot the ratios of the approximations of the norm, $|R_{i,i}|$, $\|R(i,:)\|_2$ and $\tilde{\sigma}_1$, to the exact norm.

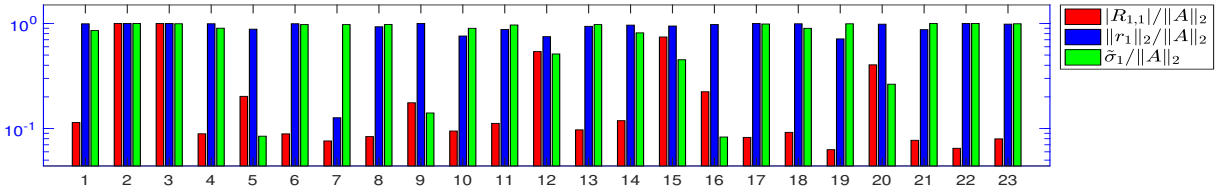


Figure 9: Ratios of the approximated matrix norm by QRCP and AGC to the exact norm, we compare $|R_{1,1}|$, $\|R(1,:)\|_2$ and $\tilde{\sigma_1}$ to $\|A\|_2$.

In Figure 10 we show AGC also provides a very good rank-one approximation, we compare the approximations of A created by QRCP and by AGC, this is

$$A \approx q_1 r_1^T, \qquad \text{and} \qquad A \approx \xi_1 = \frac{g}{\|g\|_2}\tilde{\sigma}_1 c, \quad \tilde{\sigma}_1 = \|g\|_2\sqrt{n}, \tag{107}$$

where the first is the classical QRCP rank-one approximation, see (47).

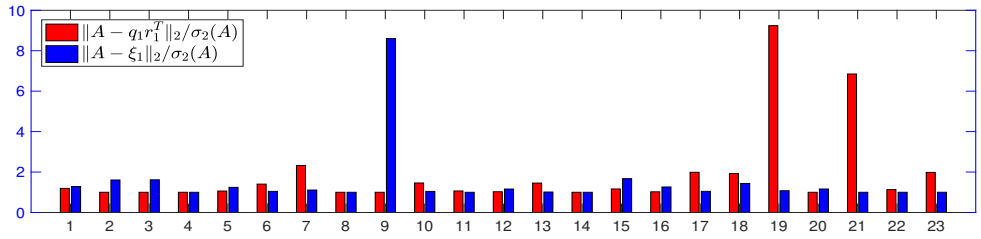
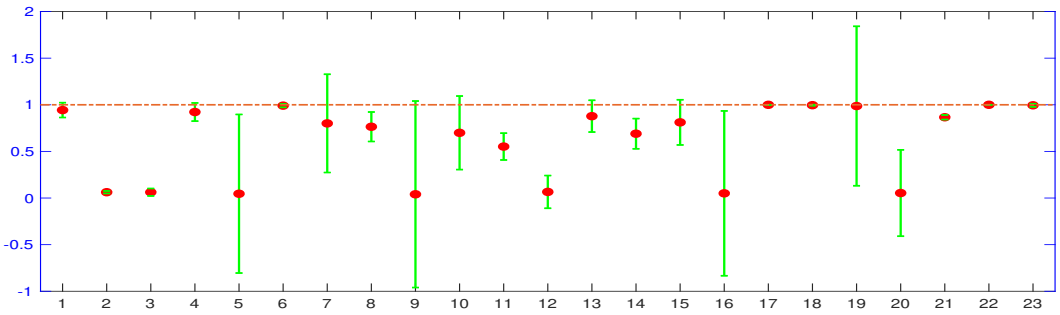


Figure 10: Ratios of the error of rank-one approximation obtained by QRCP and AGC to the optimal error.

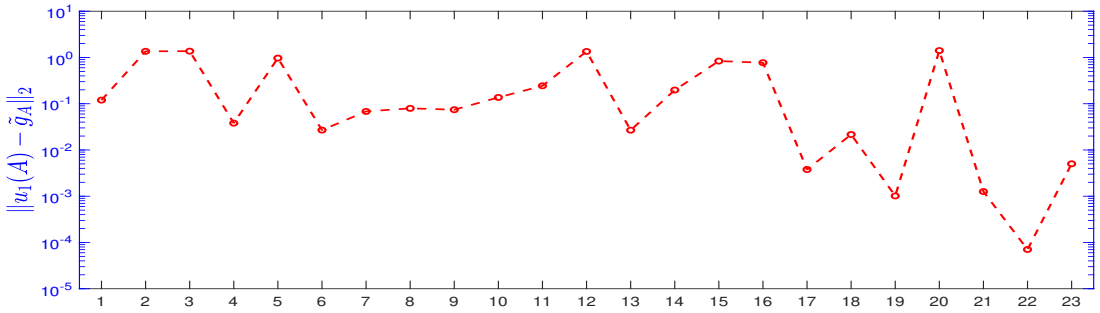
5.3 Analyzing the Correlation Coefficient

In Figure 11 we numerically study the correlation of a matrix by using the vector $\tilde{\rho}(A)$, defined in (97), and the correlation coefficient $\mathcal{G}(A)$, defined in (98), as indicators of when the best fitting lines of the matrix A coincide, and hence provide an easy way to approximate $u_1(A)$ and $v_1(A)$ according to Theorem 4.1. The matrix A stands for one of the 23 matrices from Table 1. We present three subfigures aligned in such a way that we can see that for matrices with high correlation we can approximate the first left and right singular vectors by using information of the spatial distribution of the columns and rows of A, more precisely, the gravity centers of its columns and rows.

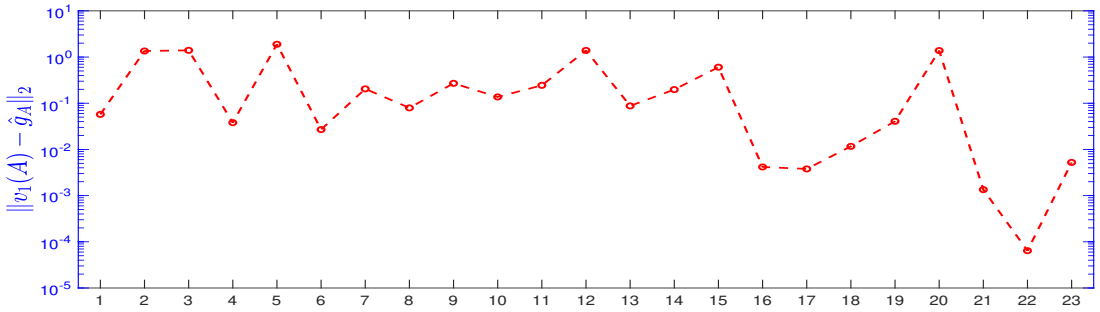
Note that, as expected, for the matrices with singular values decreasing at exponential rate, we have that the mean of the correlation vector $\tilde{\rho}(A)$ is close to 1, while the coefficient $\mathcal{G}(A)$ is close to 0, and



(a) Letting A be one of the 23 matrices from Table 1, the circle corresponds to the mean of the correlation vector $\tilde{\rho}_A$ defined in (97), and the length of the bar is equal to the correlation coefficient $\mathcal{G}(A)$ defined in (98).



(b) Error to approximate the first left singular vector $u_1(A)$ by $\tilde{g} = \tilde{s}\frac{g}{\|g\|_2}$, where g is the gravity center of the columns of A and $\tilde{s} = S(g)S(u_1(A))$.



(c) Error to approximate the first right singular vector $v_1(A)$ by $\hat{g} = \bar{s}\frac{g_t}{\|g_t\|_2}$, where g_t is the gravity center the columns of A^T and $\bar{s} = S(g_t)S(v_1(A))$.

Figure 11: Correlation vector and coefficient for the 23 matrices from Table 1, we can see that for a matrix with high correlation, we can safely approximate its left and right first singular vectors using the unit vectors in the direction of the gravity centers of its columns and rows respectively.

their singular vectors $u_1(A)$ and $v_1(A)$ can be safely approximated by the unit vectors in the directions of the gravity centers of the columns of A and A^T respectively, up to a corresponding sign. Moreover, this kind of approximation also works relatively well for some matrices with slowly decreasing singular values, such as matrices 7 and 21.

5.4 Approximation of Matrices of BEM type

Consider the three dimensional surface proposed in [1] (as shown in Figure 12) defined as $\Gamma : [0,1]\times[0,1) \to \mathbb{R}^3$, where

$$\Gamma(t,z) = \begin{bmatrix} \sqrt{z(1-z)}\cos(2\pi t) \\ \sqrt{z(1-z)}\sin(2\pi t)(2-1.5\sin(2\pi t)) \\ z \end{bmatrix}. \tag{108}$$

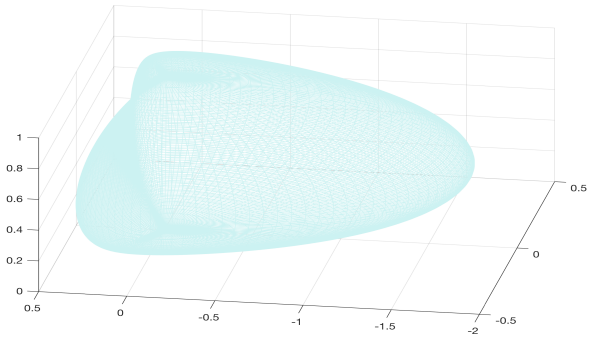


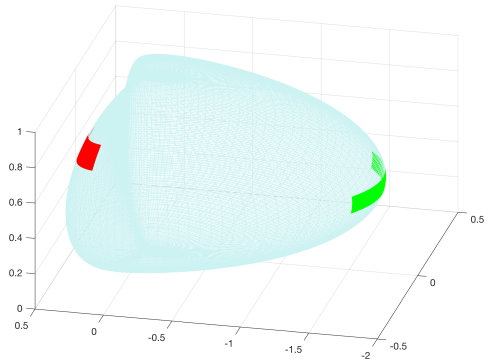
Figure 12: 3D sample surface domain defined in (108).

We choose two discrete subdomains each containing $n = 256$ points of Γ, $X = \{x_1, \cdots, x_n\}$ and $Y = \{y_1, \cdots, y_n\}$ from Γ and then construct the matrix $A \in \mathbb{R}^{n\times n}$ using the Laplacian kernel. This is, we construct the corresponding interaction matrix is given as

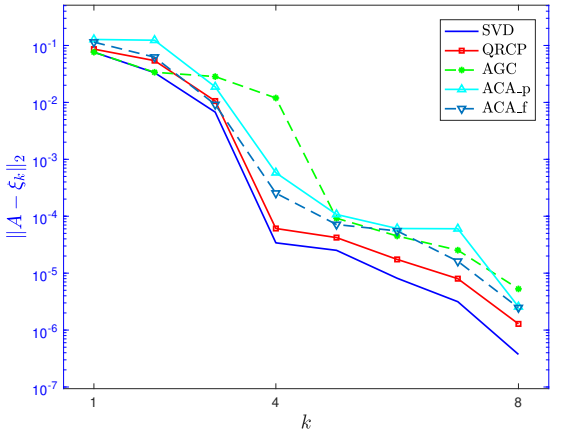
$$A_{i,j} = -\frac{1}{2\pi}\log(\|x_i - y_j\|_2) \quad \forall i,j = 1, \cdots, 256. \tag{109}$$

Figures 13 and 14 show admissible and non-admissible subdomains from which we construct, respectively, the matrices referred to as 3D-LAP-ADM and 3D-LAP-NADM in Table 1, by using the Laplacian kernel as in (109). Both figures also show the Convergence curves of the approximation error. The labels ACA_p and ACA_f stand for the adaptive cross approximation algorithm [2] with partial and full pivoting respectively. It should be notice that since for admissible domains we have that the singular values decrease exponentially [2], then the rank-one approximation by the AGC algorithm should be quite good, this is verified, and indeed its rank-one and rank-two approximations are nearly optimal, see Figure 13.

Remark 5.1. Note that in order to reduce the complexity of AGC, we can use the an approximation of the gravity center as in Section 4.3 or by directly using information from the spatial distribution of the domains in $\mathbb{R}^d$, for $d = 1, 2$ or 3. In fact, it is also possible to avoid the updates of line 8 of Algorithm AGC, this is an ongoing work of the authors.

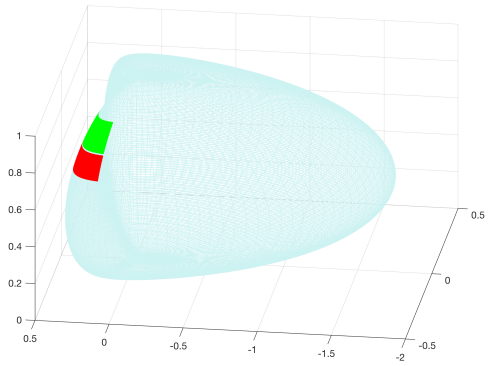


(a) Two admissible subdomains.

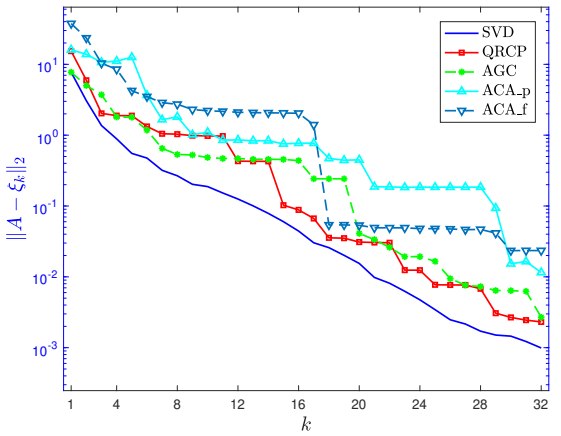


(b) Low-rank approximation error for the matrix constructed with the Laplacian kernel using the domains from the left figure.

Figure 13: Comparison of different methods to approximate a matrix corresponding to an admissible block, we can see that the rank-one and rank-two approximations are nearly optimal for the AGC algorithm.



(a) Two non-admissible subdomains.



(b) Low-rank approximation error for the matrix constructed with the Laplacian kernel using the domains from the left figure.

Figure 14: Comparison of different methods to approximate a matrix corresponding to a non-admissible block.

6 Conclusions

We have presented the concept of affine low-rank approximation for rectangular matrices, which can be interpreted geometrically as fitting the columns of the matrix into an affine subspace. We have showed how to construct an affine approximation by means of orthogonal projections and propose an algorithm named ALORA that can be adapted to any low-rank approximation algorithm. We have derived a bound for the approximation error and analyzed the cases where this approach might be advantageous by means of a correlation coefficient that we define in order to understand the geometrical structure of a matrix by seeing its columns as points of a high-dimensional space. By looking for matrices with high correlation, in the sense of our definitions, we encountered the case of matrices with exponentially decreasing singular

values for which we propose a heuristic algorithm named AGC, which also provides a fast approximation of the matrix norm.

We have constructed affine low-rank approximations using ALORA with the classical QRCP and subspace iteration algorithms. For the former, we have presented a detailed analysis of the pivoting techniques and provided a bound for the case when an arbitrary pivoting technique is used. For the latter, we have proved a result on the convergence of singular vectors, showing a bound that is in agreement with the one for convergence of singular values proved recently. The numerical experiments performed on a set of challenging matrices, showed that an affine low-rank approach can increase, in many cases, the accuracy of QRCP and subspace iteration. And although, we have only tested with sequential implementations, it can be expected that an affine approximation can also improve the accuracy of parallel algorithms for low-rank approximation. The algorithm AGC was used to approximate the norm of the test matrices, and even though most of them do not have singular values decreasing at exponential rate, we got a very good approximation of their norm. We have also tested AGC on matrices arising from the pairwise interaction of points from admissible and non-admissible 3D domains, AGC shows good accuracy to compute their small-rank approximations, however with non-linear cost.

Acknowledgement. The authors' work was supported by the NLAFET project as part of European Union's Horizon 2020 research and innovation program under grant 671633. XC acknowledges support of the French National Research Agency (ANR) contract ANR-15-CE23-0017-01 (project NonlocalDD).

A Supplemental theory

A.1 Best Fitting Line Analysis

Consider a matrix $A \in \mathbb{R}^{m\times n}$, we use the notation $A := [a_j]$, where a_j is its j-th column. By considering the vectors a_j as points on the space $\mathbb{R}^m$, we are interested in the problem of finding the line that fits the best to all these points, we write this line as

$$\mathcal{L}_A(\tau) = w + \tau u, \quad \forall \tau \in \mathbb{R}, \tag{110}$$

where $w, u \in \mathbb{R}^m$ and u is unitary.

In order to find $\mathcal{L}_A$, let us write the n points as $a_j = w + \rho_j u + \delta_j u_{\perp j}$, where $\rho_j = u^T(a_j - v)$ and $u_{\perp j}$ is a unit vector perpendicular to u with an appropriate coefficient δ_j. Also define $y_j := a_j - w$ and its corresponding matrix $Y := [y_j] \in \mathbb{R}^{m\times n}$.

Next, we write the error as a functional, depending on w and u, which measures sum of the squared distances from a_j to $\mathcal{L}_A$, for all $j = 1, \cdots, n$. This is,

$$E(w,u) = \sum_{j=1}^{n} \delta_j^2 = \sum_{j=1}^{n} \|y_j - \rho_j u\|_2^2 = \sum_{j=1}^{n} y_j^T (I - uu^T) y_j. \tag{111}$$

Existence of the solution
Fist, to find u that minimizes E, let us rewrite (111) as

$$E(w,u) = u^t \underbrace{\sum_{j=1}^{n} \left((y_j^t y_j) I - y_j y_j^T \right)}_{X} u. \tag{112}$$

Then, it is clear that E attains its minimum when u corresponds to the eigenvector associated to the smallest eigenvalue of X or, equivalently, to the greatest eigenvalue of $C := \sum_{j=1}^{n} y_j y_j^T = YY^T \in \mathbb{R}^{m\times m}$. Hence, the first singular vector of Y is a solution for u, this is

$$u = u_1(Y). \tag{113}$$

Next, in order to find w, simply set the derivative of E with respect to w equal to zero, this is

$$\frac{\partial E}{\partial w} = -2(I - uu^T)(\sum_{j=1}^{n} y_j) = 0, \tag{114}$$

where the equality trivially holds when $\sum_{j=1}^{n} y_j = 0$, or equivalently when

$$w = \frac{1}{n}\sum_{j=1}^{n} a_j =: g, \tag{115}$$

where where g is known as the gravity center of the matrix A.

Uniqueness of the solution
Clearly the choice of w is not unique, since the pair $(w + \theta u, u)$, for all $\theta \in \mathbb{R}$, also defines the same line $\mathcal{L}_A$ as the pair (w, u). Hence, we set $w = g$.

It is much more interesting to analyze if the solution for u is unique. For this case, we have that u is the eigenvector corresponding to the largest eigenvalue of $C = YY^T$, named λ_1. Then, $E(w, u)$ attains a minimum if and only if $u = u_1(Y)$, provided λ_1 has algebraic multiplicity equal to 1, since its geometric multiplicity is also going to be 1 (see e.g. [38, Sec.1]). Equivalently, the solution $u = u_1(Y)$ is unique provided $\sigma_1(Y) \neq \sigma_2(Y)$. This analysis is more general than the one made for the total least-square problem in [30, Thm. 5].

References

[1] M. Bebendorf. Approximation of boundary element matrices. *Numerische Mathematik*, 86(4):565–589, October 2000.

[2] M. Bebendorf. *Hierarchical Matrices*. Springer, Leipzig, Germany, 2008.

[3] C. Bischof. A parallel QR factorization algorithm with controlled local pivoting. *SIAM J. Matrix Anal. Appl.*, 12, 1991.

[4] C. Boutsidis, M. Mahoney, and P. Drineas. An improved approximation algorithm for the column subset selection problem. *Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 968–977, 2009.

[5] A. Çivril and M. Magdon-Ismail. Exponential inapproximability of selecting a maximum volume sub-matrix. *Algorithmica*, 65(1):159–176, January 2013.

[6] J. W. Demmel. *Applied Numerical Linear Algebra*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1997.

[7] J. W. Demmel, L. Grigori, M. Gu, and H. Xiang. Communication avoiding rank revealing QR factorization with column pivoting. *SIAM J. Matrix Anal. Appl.*, 2015.

[8] Z. Drmač and Z. Bujanović. On the failure of rank-revealing QR factorization software – a case study. *ACM Trans. Math. Softw.*, 35(2):12:1–12:28, July 2008.

[9] Z. Drmač and K. Veselić. New fast and accurate Jacobi SVD algorithm. I. *SIAM J. Matrix Anal. Appl.*, 29(4):1322–1342, 2008.

[10] Z. Drmač and K. Veselić. New fast and accurate Jacobi SVD algorithm. II. *SIAM J. Matrix Anal. Appl.*, 29(4):1343–1362, 2008.

[11] J. Duersch and M. Gu. Randomized QR with column pivoting. *SIAM J. Sci. Comput.*, 39(4):C263–C291, 2017.

[12] G. Eckart and Y. G. The approximation of one matrix by another of lower rank. *Psychometrica*, (1):211–218, 1936.

[13] A. Edelman. Eigenvalues and condition numbers of random matrices. *SIAM J. Matrix Anal. Appl.*, 9(4):543–560, Dec. 1988.

[14] W. Fong and E. Darve. The black-box fast multipole method. *Journal of Computational Physics.*, 228:8712–8725, 2009.

[15] A. Frieze, R. Kannan, and S. Vempala. Fast monte-carlo algorithms for finding low-rank approximations. *J. ACM*, 51(6):1025–1041, Nov. 2004.

[16] G. Golub, V. Klema, and S. G.W. Rank degeneracy and least squares problems. *Tech. Report TR-456, Department of Computer Science, University of Maryland, College Park*, MD, 1976.

[17] G. Golub and C. Van Loan. *Matrix Computations*. Jonhs Hopkins University Press, Baltimore, 3rd edition, 1996.

[18] S. Goreinov and E. Tyrtyshnikov. The maximal-volume concept in approximation by low-rank matrices. *Contemporary Mathematics*, (280):47–51, 2001.

[19] L. Grigori, S. Cayrols, and J. W. Demmel. Low rank approximation of a sparse matrix based on LU factorization with column and row tournament pivoting. Research Report RR-8910, INRIA, 2016.

[20] M. Gu. Subspace iteration randomization and singular value problems. *SIAM J. Sci. Comput.*, 2015.

[21] M. Gu and S. Eisenstat. Efficient algorithms for computing a strong rank-revealing QR factorization. *SIAM J. Matrix Anal. Appl.*, 17(4):848–869, 1996.

[22] N. Halko, P. G. Martinsson, and J. A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev.*, 53(2):217–288, May 2011.

[23] P. Hansen. *Regularization tools version 4.1 for matlab 7.3.* http://www.imm.dtu.dk/~pcha/Regutools. Accessed 10 Mar 2018.

[24] R. Horn and C. Johnson. *Topics in Matrix Analysis*. Cambridge University Press, New York, USA, 1991.

[25] A. Householder. Unitary triangularization of a nonsymmetric matrix. *J. ACM*, 5(4):339–342, 1958.

[26] D. Huckaby and C. T.F. Stewart's pivoted QLP decomposition for low-rank matrices. *Numer. Linear Algebra Appl.*, 12(4):153–159, 2005.

[27] W. Kahan. Numerical linear algebra. *Canadian Math. Bull*, (9):757–801, 1966.

[28] R. Larsen. Lanczos bidiagonalization with partial reorthogonalization. Technical report, Department of Computer Science, Aarhus University, Aarhus, Denmark, 1998. Also available online from http://soi.stanford.edu/~rmunk/PROPACK. Accessed 10 Mar 2018.

[29] R. Lehoucq, D. Sorensen, and C. Yang. *ARPACK Users' Guide*. Society for Industrial and Applied Mathematics., Philadelphia, 1998.

[30] I. Markovsky and S. Van Huffel. Overview of total least-squares methods. *Signal Process.*, 87(10):2283–2302, Oct. 2007.

[31] P. Martinsson. Randomized methods for matrix computations and analysis of high dimensional data. *arXiv:1607.01649*, 2016.

[32] P. Martinsson, V. Rokhlin, and M. Tygert. A randomized algorithm for the approximation of matrices. *Technical Report Yale CS research report YALEU/DCS/RR-1361, Yale University, Computer Science Department*, 2006.

[33] L. Mirsky. Symmetric gauge functions and unitarily invariant norms. *Quart. J. Math. Oxford Ser.*, 11(2):50–59, 1960.

[34] S. O'Rourke, V. Vu, and K. Wang. Eigenvectors of random matrices: A survey. *J. Comb. Theory Ser. A*, 144:361–442, 2016.

[35] N. A. Ozdemir and J.-F. Lee. A low-rank IE-QR algorithm for matrix compression in volume integral equations. *IEEE Transactions on Magnetics*, 40(2):1017–1020, 2004.

[36] C.-T. Pan. On the existence and computation of rank-revealing LU factorizations. *Linear Algebra and its Applications*, 316(1):199 – 222, 2000.

[37] C.-T. Pan and P. Tang. Bounds on singular values revealed by QR factorizations. *BIT Numerical Mathematics*, 39(4):740–756, December 1999.

[38] A. Quarteroni, R. Sacco, and F. Saleri. *Numerical Mathematics (Texts in Applied Mathematics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[39] M. Rudelson. Invertibility of random matrices: norm of the inverse. *Annals of Mathematics*, 168:575–600, 2008.

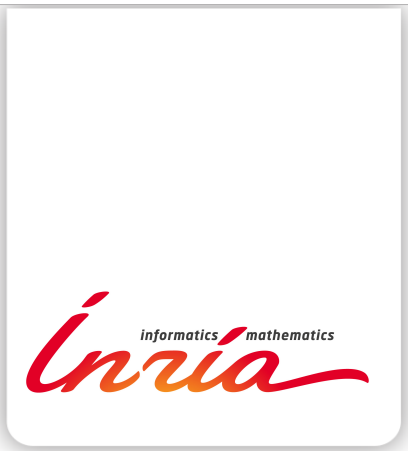
[40] P. Schneider and D. Eberly. *Geometric Tools for Computer Graphics*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2003.

[41] G. W. Stewart. The QLP approximation to the singular value decomposition. *SIAM J. Sci. Comput.*, 20(4):1336–1348, Feb. 1999.

[42] S. Szarek. Condition numbers of random matrices. *J. Complexity*, 7:131–149, 1991.

[43] T. Tao and V. Vu. Random matrices: Universal properties of eigenvectors. *Random Matrices Theory Appl.*, 1(1):1150001, 2012.

[44] S. Voronin and P. Martinsson. RSVDPACK: An implementation of randomized algorithms for computing the singular value, interpolative, and CUR decompositions of matrices on multi-core and GPU architectures. *arXiv:1502.05366*, 2015.



RESEARCH CENTRE
PARIS

2 rue Simone Iff
CS 42112 - 75589 Paris Cedex 12

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399